



TECHNISCHE
UNIVERSITÄT
DARMSTADT

ULB

Stimulation eines Radar-Objekterkennungs- und Trackingverfahren mit synthetischen Daten

Zaman, Zahra

(2020)

DOI (TUprints): <https://doi.org/10.25534/tuprints-00013502>

Lizenz:



CC-BY-NC-ND 4.0 International - Creative Commons, Namensnennung, nicht kommerziell, keine Bearbeitung

Publikationstyp: Masterarbeit

Fachbereich: 16 Fachbereich Maschinenbau

Quelle des Originals: <https://tuprints.ulb.tu-darmstadt.de/13502>

Stimulation eines Radar Objekterkennungs- und Trackingverfahren mit synthetischen Daten

Masterthesis Nr. 761/19

Bearbeiter: Zahra Zaman | 2047223

Betreuer: Martin Holder, M. Sc.

Co-Betreuer: Clemens Linnhoff, M. Sc., Philipp Rosenberger, M. Sc.



TECHNISCHE
UNIVERSITÄT
DARMSTADT



FAHRZEUGTECHNIK
TU DARMSTADT

Zahra Zaman

Matrikelnummer: 2047223

Studiengang: Mechanical & Process Engineering, M. Sc.

Masterthesis Nr. 761/159

Thema: Stimulation eines Radar Objekterkennungs- und Trackingverfahren mit synthetischen Daten

Eingereicht: 02. Juni 2020

Technische Universität Darmstadt

Fachgebiet Fahrzeugtechnik

Prof. Dr. rer. nat. Hermann Winner

Otto-Berndt-Straße 2

64287 Darmstadt

Dieses Dokument wird bereitgestellt von TUpriints – Publikationsservice der TU Darmstadt.

<https://tuprints.ulb.tu-darmstadt.de/>

Bitte verweisen sie auf:

URN: <urn:nbn:de:tuda-tuprints-135025>

URI: <https://tuprints.ulb.tu-darmstadt.de/id/eprint/13502>

Lizenz: CC-BY-NC-ND 4.0 International

<https://creativecommons.org/licenses/by-nc-nd/4.0/>

Erklärung

Erklärung zur Abschlussarbeit gemäß § 23 Abs. 7 APB der TU Darmstadt

Hiermit versichere ich, Zahra Zaman, die vorliegende Master-Thesis gemäß § 22 Abs. 7 APB der TU Darmstadt ohne Hilfe Dritter und nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die Quellen entnommen wurden, sind als solche kenntlich gemacht worden. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Mir ist bekannt, dass im Falle eines Plagiats (§38 Abs.2 APB) ein Täuschungsversuch vorliegt, der dazu führt, dass die Arbeit mit 5,0 bewertet und damit ein Prüfungsversuch verbraucht wird. Abschlussarbeiten dürfen nur einmal wiederholt werden.

Bei einer Thesis des Fachbereichs Architektur entspricht die eingereichte elektronische Fassung dem vorgestellten Modell und den vorgelegten Plänen.

English translation for information purposes only:

Thesis Statement pursuant to § 23 paragraph 7 of APB TU Darmstadt

I herewith formally declare that I, Zahra Zaman, have written the submitted thesis independently. I did not use any outside support except for the quoted literature and other sources mentioned in the paper. I clearly marked and separately listed all of the literature and all of the other sources, which I employed when producing this academic work, either literally or in content. This thesis has not been handed in or published before in the same or similar form.

I am aware, that in case of an attempt at deception based on plagiarism (§38 Abs. 2 APB), the thesis would be graded with 5,0 and counted as one failed examination attempt. The thesis may only be repeated once.

For a thesis of the Department of Architecture, the submitted electronic version corresponds to the presented model and the submitted architectural plans.

Darmstadt, 02. Juni 2020

Zusammenfassung

Obwohl heute noch keine automatisierten Fahrzeuge flächendeckend im Straßenverkehr zu finden sind, werden moderne Fahrzeuge vermehrt mit Fahrerassistenzsystemen ausgestattet, um bereits so eine Steigerung der Sicherheit, aber auch des Komforts zu erreichen. Bis zu seiner Einführung trifft das automatisierte Fahren auf Hürden, welche insbesondere die Freigabe der automatisierten Fahrzeuge betreffen. Die Freigabe benötigt, ebenfalls zur Absicherung der Automobilhersteller, einen umfassenden Sicherheitsnachweis. Dieser Nachweis erfordert Feldversuche und deren anschließenden Auswertung, sodass Aussagen zum Sicherheitspotenzial einer automatisierten Fahrt getroffen werden können. Um den Aufwand der Realfahrten, welcher durch Feldversuche verursacht wird, zu senken, wird ein Teil der Realfahrten durch Sicherheitsnachweise in Simulationsumgebungen ersetzt. Diese Simulationsumgebungen bilden nicht nur das Fahrzeug und die Umgebung virtuell ab, sondern modellieren auch die erforderlichen Sensoren. Aus diesem Grund wurde am Fachgebiet für Fahrzeugtechnik an der Technischen Universität Darmstadt ein Radarsensormodell entwickelt und in der Simulationssoftware Virtual Test Drive von Vires implementiert. Dieses Modell erzeugt mithilfe eines Raytracing-Verfahrens synthetische Rohdaten und extrahiert hieraus einzelne Detektionen. Die zeitliche Nachverfolgung dieser Einzeldetektionen, dem sogenannten Tracking, ermöglicht, Detektionen zu Objekten zusammenzufassen. Beim FZD steht der Radarsensor eines Automobilzulieferers zur Verfügung, welcher allerdings keinen Einblick in die genaue Funktionsweise des Tracking-Moduls erlaubt, sodass eine Anpassung dieses Moduls zur zusätzlichen Verarbeitung von synthetischen Rohdaten nicht möglich ist. Ziel dieser Masterthesis ist es daher, einen Algorithmus zur Radar-Objekterkennung und Tracking zu implementieren, welcher in der Lage ist sowohl Mess- als auch Simulationsdaten derart zu verarbeiten, dass eine Objektliste ausgegeben wird.

Im theoretischen Teil der Arbeit werden zunächst die Grundlagen der Radarsensorik zusammengefasst (2) und anschließend eine Anforderungsanalyse der geplanten Implementierung (3) vorgenommen. Darauf folgend wird der Stand der Technik zu Tracking-Algorithmen vorgestellt sowie deren Anwendung auf die vorhandenen Radardaten analysiert, was eine Auswahl der zu implementierenden Methoden ermöglicht (4). Ausgehend davon werden die Systemarchitektur (5) beschrieben und anschließend die implementierten Methoden (6) erläutert. Die Evaluierung des Tracking-Algorithmus (7) erfolgt mit einem Vergleich zur Industrielösung und einer Untersuchung der Übertragbarkeit der Ergebnisse zwischen Simulations- und Messdaten. Dabei wird insbesondere auf Unterschiede in der Positions-, Geschwindigkeits- und Gierwinkelschätzung eingegangen. Dabei wird deutlich, dass der implementierte Tracking-Algorithmus teilweise, insbesondere in der Gierwinkelschätzung, bessere Ergebnisse erzielt. Die Arbeit schließt mit einer Ergebnisdiskussion und einem Ausblick auf mögliche Weiterentwicklungen (8) ab.

Inhaltsverzeichnis

Zusammenfassung.....	I
Inhaltsverzeichnis.....	II
Formelzeichen und Indizes	IV
Abkürzungsverzeichnis.....	VI
Abbildungsverzeichnis.....	VII
Tabellenverzeichnis	IX
1 Einleitung	1
1.1 Motivation.....	1
1.2 Konkretisierung der Aufgabenstellung	3
1.3 Methodik des Vorgehens	3
2 Grundlagen.....	5
2.1 Radarsensorik.....	5
2.1.1 Aufgaben.....	5
2.1.2 Aufbau.....	6
2.1.3 Messgrößen und Messprinzipien	6
2.1.4 Signalverarbeitung	11
2.1.5 Mehrdeutigkeiten	14
2.2 Radarmodell.....	16
3 Anforderungsanalyse	18
3.1 Definition	18
3.1.1 Fahrzeugkoordinatensystem.....	18
3.1.2 Sensorkoordinatensysteme.....	18
3.1.3 Objekt.....	19
3.1.4 Tracking	19
3.2 Ableitung der Anforderungen	20
4 Tracking-Algorithmen	21
4.1 Einführung in das Tracking.....	21
4.2 Tracking-Arten.....	24
4.3 Systemarchitektur.....	25
4.4 Clustering	26
4.5 Datenassoziation	29
4.5.1 Probabilistic Data Association und Erweiterungen	29
4.5.2 Multiple-Hypothesis Tracking	30
4.5.3 Nearest Neighbor und Global Nearest Neighbor	31
4.5.4 Random Finite Sets	32
4.6 Track-Management	32
4.7 Prädiktion und Innovation.....	34

4.7.1	Bewegungsmodelle zur Prädiktion	34
4.7.2	Filtermethoden	35
4.8	Zusammenfassung.....	38
5	Systemarchitektur.....	39
6	Implementierung	43
6.1	Filterung der Detektionen	43
6.2	Umrechnung der Bin-Daten in radiale und kartesische Koordinaten	45
6.3	Clustering	46
6.4	Datenassoziation	49
6.5	Track-Management	50
6.6	Prädiktion und Filterung	52
6.6.1	Initiale Prädiktion und Filterung	52
6.6.2	Prädiktion und Filterung mit dem Extended-Kalman-Filter	53
6.6.3	Eigenbewegungskompensation	57
6.6.4	Schätzung der kartesischen Geschwindigkeiten	57
6.6.5	Schätzung der Bounding-Box	61
6.7	Löschung von Duplikaten	63
6.8	Zusammenfassung.....	63
7	Evaluierung	65
7.1	Leistungsbewertung von Tracking-Algorithmen	65
7.2	Vergleich der Leistungsfähigkeit mit der Industrielösung.....	66
7.2.1	Initiierung und Löschung von Objekten	66
7.2.2	Geschwindigkeitsschätzung	70
7.2.3	Positionsschätzung	75
7.2.4	Gierwinkelschätzung.....	80
7.2.5	Bounding-Box-Schätzung	82
7.2.6	Trennfähigkeit	85
7.2.7	Kontinuität	87
7.3	Vergleich der Ergebnisse zwischen Messung und Simulation	88
7.4	Zusammenfassung.....	94
8	Fazit und Ausblick	97
Anhang A: Parameterdatei		100
Anhang B: Visualisierung		101
Anhang C: Soll- und Ist-Zeitplan.....		102
Literaturverzeichnis		104

Formelzeichen und Indizes

Lateinische Buchstaben:

Zeichen	Einheit	Bezeichnung
a	m/s^2	Beschleunigung
A	m	Amplitude
c	m/s	Lichtgeschwindigkeit
d	m	Euklidische Distanz
f	Hz	Frequenz
$f(\cdot)$	-	Zustandsübergangsfunktion
F	-	Zustandsübergangsmatrix
G	-	Antennengewinn
I	m^2 und m^2/s^2	Innovation
k	dB/km	Absorption
m	Hz/s	Steigung einer Frequenzrampe
n	-	Anzahl
P	W	Leistung
$p(\cdot)$	-	Wahrscheinlichkeit
r	m	Radiale Entfernung
\dot{r}	m/s	Radiale Relativgeschwindigkeit (Dopplergeschwindigkeit)
\mathbf{S}	-	Zustandsvektor (Unterschiedliche Einheiten je Komponente)
T_M	ms	Messzeit
T	s	Zykluszeit
v	m/s	Geschwindigkeit
\mathbf{v}	-	Modellrauschen (Unterschiedliche Einheiten je Komponente)
V_{mp}	-	Schüttelfaktor
\mathbf{w}	-	Sensorrauschen (Unterschiedliche Einheiten je Komponente)
x	m	Entfernung in x-Richtung
y	m	Entfernung in y-Richtung
\mathbf{y}	-	Messvektor (Unterschiedliche Einheiten je Komponente)
t	-	Zyklus
Δt	s	Laufzeit (time of flight)

Griechische Buchstaben:

Zeichen	Einheit	Bezeichnung
ϑ	rad	Elevationswinkel
α	rad	Azimutwinkel
Γ	-	Normierter Abstand zwischen den Antennen



σ	- / m ²	Nach Kontext: Standardabweichung (Einheit der Bezugsgröße) oder Radarquerschnitt
λ	m	Wellenlänge
ϕ	rad	Phase
Δ	-	Differenz (Einheit der Bezugsgröße)
φ	rad	Gierwinkel
$\dot{\phi}$	rad/s	Gierrate

Indizes:

Zeichen	Bezeichnung
0	Ausgangsgröße
Bin	Binwert der Bezugsgröße
comp	abstandskompensiert
clust	Cluster
D	Direktiv
dop	Doppler
E	Empfänger
ego	Egofahrzeug
hyp	Hypothese
k	Bis zum Zeitpunkt k
log	logarithmiert
max	Maximal
mess	Messdaten
min	Minimal
obj	Objekt
pos	Position
pred	Prädiktion
r	Entfernung
\dot{r}	Radiale Relativgeschwindigkeit
R	Rampen
real	Real
rel	Relativ
s	Abtastung
S	Sender
sen	Sensor
total	Gesamt
track	Track
vel	Geschwindigkeit
x	x-Komponente (Fahrzeugkoordinatensystem)
y	y-Komponente (Fahrzeugkoordinatensystem)

Abkürzungsverzeichnis

ACC	Adaptive Cruise Control
ADMA	Automotive Dynamic Motion Analyzer
CFAR	Constant False-Alarm Rate
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
EKF	Extended Kalman Filter
EOT	Extended Object Tracking
FFT	Fast Fourier Transformation
FZD	Fahrzeugtechnik Darmstadt
GNN	Global Nearest Neighbor
GOT	Group Object Tracking
IMM	Interacting Multiple Model
IVE	Instantaneous Velocity Estimation
JIPDA	Joint Integrated Probabilistic Data Association
JPDA	Joint Probabilistic Data Association
MHT	Multiple-Hypothesis Tracking
MOT	Multi-Object Tracking
NN	Nearest Neighbor
PDA	Probabilistic Data Association
POT	Point Object Tracking
RCS	Radar Cross Section
RFS	Random Finite Sets
SOT	Single-Object Tracking
TUD	Technische Universität Darmstadt
UKF	Unscented Kalman Filter
VTD	Virtual Test Drive

Abbildungsverzeichnis

Abbildung 1-1: Darstellung der Methodik des Vorgehens	3
Abbildung 2-1: Prinzip des Pulsradars.....	6
Abbildung 2-2: Idealisierte Darstellung einer Amplituden- und Frequenzmodulation.; links: Aufmodulation eines Pulssignals; rechts: Aufmodulation eines Sinussignals ^{15a}	7
Abbildung 2-3: Frequenzrampen eines Pulskompressionsradars	8
Abbildung 2-4: Signlverarbeitungskette eines Radarsensors.....	11
Abbildung 2-5: Überführung des Empfangssignals aus dem Zeitbereich (links) in den Frequenzbereich (rechts)	12
Abbildung 2-6: Dreidimensionaler FFT mit Bindarstellung.....	13
Abbildung 2-7: Erweitern des eindeutigen FFT Wertebereichs und die damit verbundene Mehrdeutigkeit	15
Abbildung 2-8: Sichtbereich des Radarsensors und des Radarmodells.....	16
Abbildung 2-9: Backward-Raytracing	17
Abbildung 3-1: Fahrzeugkoordinatensystem	18
Abbildung 3-2: Polares (links) und kartesisches (rechts) Sensorkoordinatensystem	19
Abbildung 4-1: Systemarchitektur eines Tracking-Algorithmus.....	25
Abbildung 4-2: Reflexionszentren an einer Fahrzeugkarosserie ⁵⁸	27
Abbildung 4-3: Unterschiedliche Reflexionsdichte eines Fahrzeugs in Abhängigkeit der Entfernung in kartesischen Koordinaten (oben) und im Azimutwinkel-Entfernungs-Gitter (unten)	28
Abbildung 4-4: Mögliches Szenario zur Datenassoziation mit prädictierten Objektzuständen und Messdaten.....	29
Abbildung 4-5: Unterschiedliche Ergebnisse der Datenassoziation nach den Verfahren des Nearest Neighbor (links) und Global Nearest Neighbor (rechts).....	31
Abbildung 4-6: Mögliche Zustandsänderung von verfolgten Objekten ^a	33
Abbildung 5-1: Systemarchitektur des entwickelten Algorithmus	39
Abbildung 6-1: Szenario zur Messung der Signalleistung.....	44
Abbildung 6-2: Vergleich der abstandskompensierten Signalleistung zur reinen Signalleistung über die Entfernung	45
Abbildung 6-3: Euklidische Distanzen zwischen Detektionen und deren Dopplerbins	47
Abbildung 6-4: Finale Clusterzuordnung	48
Abbildung 6-5: Statusübergänge der Objekttracks	51
Abbildung 6-6: Resultierender Suchbereich der initialen Prädiktion	53
Abbildung 6-7: Darstellung der Beziehung zwischen Dopplergeschwindigkeit und den kartesischen Komponenten der relativen Geschwindigkeit	58
Abbildung 6-8: Gierwinkel φ aus Positionsänderung	60
Abbildung 6-9: Ablaufdiagramm zur Wahl der Berechnungsmethode	61
Abbildung 6-10: Beispielhafte Schätzung der Bounding-Box	62
Abbildung 7-1: Vergleich der Zeitspannen bis zur Objekterkennung beider Tracker.....	67
Abbildung 7-2: Vergleich der Zeitspannen bis zur Objektlöschung beider Tracker	69
Abbildung 7-3: Vergleich der geschätzten Relativgeschwindigkeiten zwischen implementierten Tracker und der Industrielösung für Szenario 1	71
Abbildung 7-4: Vergleich der geschätzten Relativgeschwindigkeiten zwischen implementierten Tracker und der Industrielösung für Szenario 2	72
Abbildung 7-5: Vergleich der geschätzten Relativgeschwindigkeiten zwischen implementierten Tracker und der Industrielösung für Szenario 3	73

Abbildung 7-6: Fehler in der longitudinalen Geschwindigkeitsschätzung in Abhängigkeit der Beschleunigung für Szenario 3	73
Abbildung 7-7: Vergleich der geschätzten lateralen Relativgeschwindigkeiten zwischen implementierten Tracker und der Industrielösung für Szenario 2.....	74
Abbildung 7-8: Vergleich der Differenzen in der longitudinalen Positionsschätzung für Szenario 1 bis 3	76
Abbildung 7-9: Vergleich der Differenzen in der lateralen Positionsschätzung für Szenario 1 bis 378	
Abbildung 7-10: Vergleich der Positionsschätzung für eine Kurvenfahrt und Darstellung der lateralen Differenz.....	79
Abbildung 7-11: Vergleich des geschätzten Gierwinkels φ zwischen implementierten Tracker und der Industrielösung für eine Slalomfahrt.....	81
Abbildung 7-12: Vergleich der Fehler in der Gierwinkelschätzung $\Delta\varphi$ in Abhängigkeit der Gierrate φ	81
Abbildung 7-13: Schätzung der kartesischen Entfernungen zweier Zielfahrzeuge zur Bewertung der Bounding-Box-Schätzung	83
Abbildung 7-14: Vergleich der Gierwinkelschätzung der statischen Zielfahrzeuge aufgetragen über die relative Entfernung in x-Richtung.....	84
Abbildung 7-15: Beispielhafter Vergleich der geschätzten Bounding-Boxen beider Tracker	85
Abbildung 7-16: Longitudinale Geschwindigkeitsschätzung zur Bewertung der Trennfähigkeit der Tracker	86
Abbildung 7-17: Positionsschätzung zweier Fahrzeuge zur Bewertung der Trennfähigkeit der Tracker	87
Abbildung 7-18: Szenario zur Evaluierung der Track-Kontinuität	87
Abbildung 7-19: Vergleich der Trackfragmentierung eines verfolgten Objekts anhand der longitudinalen Geschwindigkeitsschätzung	88
Abbildung 7-20: Vergleich der Geschwindigkeitsschätzung zu Mess- sowie Simulationsdaten.....	90
Abbildung 7-21: Vergleich der Gierwinkelschätzung zu Mess- sowie Simulationsdaten.....	91
Abbildung 7-22: Vergleich der Positionsschätzung zu Mess- sowie Simulationsdaten.....	92
Abbildung 7-23: Vergleich der Trennfähigkeit mit Variation der lateralen Distanz zu Mess- sowie Simulationsdaten	93
Abbildung 7-24: Vergleich der Trennfähigkeit mit Variation der Relativgeschwindigkeit zu Mess- sowie Simulationsdaten.....	93
Abbildung 7-25: Vergleich der Dopplergeschwindigkeiten des realen Radarsensor sowie des Radarsensormodells	94

Tabellenverzeichnis

Tabelle 1: Anforderungen an den zu implementierenden Tracking-Algorithmus.....	20
Tabelle 2: Potenzielle Dopplergeschwindigkeiten der Detektion 3 im Vergleich zu errechneten Dopplergeschwindigkeiten der restlichen Detektionen 1 und 4.....	48
Tabelle 3: Genutzte Variablen für Statusübergänge	50
Tabelle 4: Gewählte Wertebereiche und finale Werte der Parameter des EKF	57
Tabelle 5: Szenarienverläufe zur Bewertung der Track-Initiierung	67
Tabelle 6: Szenarienverläufe zur Bewertung der Track-Löschung	68
Tabelle 7: Szenarien zur Auswertung der Geschwindigkeitschätzung	70
Tabelle 8: Genutzte Parameter innerhalb des Tracking-Algorithmus wie Schwellenwerte oder Mindestmaße	100

1 Einleitung

Die Vision des unfallfreien Fahrens ist gegenwärtig ein wesentlicher Antrieb für die fortschreitende Automatisierung von Fahrzeugen.¹ Diese Automatisierung beinhaltet unter anderem den Einsatz von Objekt-Tracking-Systemen in modernen Fahrzeugen. Im folgenden Kapitel wird daher zur Einführung in die Thematik der Radar-Objekterkennung und des Trackings die Motivation (1.1) zur Forschung auf diesem Gebiet dargelegt. Durch die Konkretisierung der Aufgabenstellung (1.2) wird anschließend der Rahmen dieser Masterthesis erläutert, sodass darauffolgend auf die Methodik des Vorgehens (1.3) zur Bearbeitung der Aufgabenstellung eingegangen wird.

1.1 Motivation

Laut der Weltgesundheitsorganisation starben im Jahr 2010 1,24 Millionen Menschen durch Verkehrsunfälle weltweit. Im Jahr 2016 stieg diese Zahl sogar auf 1,35 Millionen Menschen.² Ohne Erhöhung der Verkehrssicherheit wird dabei von einer weiteren Zunahme der Zahl der Unfalltoten ausgegangen.³ Da Fehler menschlicher FahrerInnen den Großteil aller Verkehrsunfälle verursachen,⁴ kann eine Automatisierung von Fahrzeugen und die damit einhergehende Eliminierung des menschlichen Faktors im Straßenverkehr die Zahl der Unfalltoten um bis zu 90 Prozent senken.⁵

Neben dieser unfallvermeidenden Wirkung und einer erhöhten Sicherheit im Verkehrssystem stellen automatisierte Fahrzeuge auch weiteren Nutzen für FahrerInnen und Umwelt dar. Speziell vollautomatisierte Fahrzeuge bieten hier persönliche Nutzeffekte hinsichtlich eines neuen Fahrvergnügens. Beispielweise kann die Fahrtzeit insbesondere in Stausituationen produktiv gestaltet werden.¹ Langfristige positive Wirkungen auf die Umwelt kann automatisiertes Fahren durch eine verbesserte Straßenauslastung und sinkenden Emissionen pro Fahrzeug erzielen. Dabei wird das Potenzial einer Optimierung des Verkehrsflusses unter anderem mit einer Kapazitätserhöhung um etwa 80 Prozent auf Autobahnabschnitten mit rein automatisiertem Verkehr geschätzt.⁶

Allerdings trifft das automatisierte Fahren bis zu seiner Einführung ebenso auf Hürden. So zeigte im Jahr 2019 eine internationale Befragung von Autofahrern und Autofahrerinnen aus Deutschland, Großbritannien, den USA, China, Südkorea und Japan zwar eine prinzipielle Offenheit gegenüber dem automatisierten Fahren, jedoch wird diese Einstellung bei einer großen Zahl der Befragten von Angst begleitet, da an der sicheren Funktionsweise der Technologie gezweifelt wird. Insbesondere änderte sich nach dem Bekanntwerden von ersten Unfällen in Zusammenhang mit automatisierten Fahrzeugen die Einstellung bei 20 Prozent der Befragten ins Negative.⁷ Somit ist zur Bewältigung

¹ Maurer, M.: Motivatoren in der Forschung (2015), S. 4.

² World Health Organization: Global Status Report on Road Safety (2018), S. 4.

³ World Health Organization: Road safety laws addressing key risk factors (2013).

⁴ National Highway Traffic Security Administration: Critical Reasons for Crashes (2018), S. 2.

⁵ Chan, C.-Y.: Impacts of automated driving systems (2017), S. 214.

⁶ Friedrich, B.: Effizienzgewinn durch autonomes Fahren (2015), S. 346.

⁷ Audi AG: The Pulse of Autonomous Driving (2019), S. 18–24.

dieser Nutzerbedenken und ebenso zur Absicherung der Automobilhersteller ein Sicherheitsnachweis für automatisierte Fahrzeuge unerlässlich.

Mittels Feldversuche und der anschließenden Auswertung auf Basis der dort gesammelten Datenmengen können dabei Aussagen zur Informationsverarbeitung und somit zum Leistungs- und Sicherheitspotential einer automatisierten Fahrt getroffen werden. Aktuelle Testmethoden für Fahrer- Fahrzeug-Systeme verursachen bei der Anwendung auf fahrerlose Fahrzeuge jedoch einen überhöhten Arbeits-, Zeit- und ökonomischen Aufwand, d.h. um den speziellen Anforderungen eines fahrerlosen Fahrzeuges gerecht zu werden, müssen umfangreichere Feldversuche durchgeführt werden. Beispielsweise werden für den Sicherheitsnachweis eines Autobahnpiloten mehr als 6 Milliarden Kilometern an Testfahrten auf öffentlichen Straßen benötigt.^{8a} Da dieser Test für jede Systemmodifikation erneut durchzuführen ist und die Zahl der benötigten Testkilometer in den nächsten Jahrzehnten dabei zusätzlich steigen wird,⁹ stellt dieser Sicherheitsnachweis unter Realfahrten auf öffentlichen Straßen einen nicht vertretbaren Aufwand dar.¹⁰

Um diesen Aufwand zu senken, können Sicherheitsnachweise in Simulationsumgebungen einen Teil der Realfahrten ersetzen. Neben den ökonomischen Vorteilen bieten Tests in Simulationsumgebungen insbesondere eine höhere Reproduzierbarkeit und Variierbarkeit. Dabei ist zwischen mehreren Abstraktionsstufen zu unterscheiden. Je mehr Komponenten des Fahrzeuges und der Umgebung virtuell abgebildet werden, desto größer ist die Abstraktion. Die höchste Abstraktionsstufe stellt das Konzept Software in the Loop (SiL) dar, bei dem sowohl das Fahrzeug als auch die Umgebung virtuell modelliert werden, sodass diese Methode in frühen Entwicklungsphasen Einsatz findet. Die nächste Stufe beschreibt das Konzept Hardware in the Loop (HiL), welches die zuvor getestete Software auf der jeweiligen Hardware in einer virtuellen Umgebung prüft. Der niedrigste Abstraktionsgrad repräsentiert das Konzept Vehicle in the Loop (ViL), bei dem das Gesamtfahrzeug in einer virtuellen Umgebung getestet wird. Je nach Simulationsart ist es daher nötig, Sensoren in Form von Modellen abzubilden.^{8b}

Aus diesem Grund wurde am Fachgebiet für Fahrzeugtechnik (FZD) an der Technischen Universität Darmstadt (TUD) ein Radarsensormodell entwickelt und in der Simulationssoftware Virtual Test Drive (VTD) von Vires implementiert. Dieses Modell erzeugt synthetische Rohdaten und extrahiert hieraus einzelne Detektionen. Über eine zeitliche Nachverfolgung dieser Einzeldetektionen, dem sogenannten Tracking, ist es möglich, Detektionen zu Objekten zusammenzufassen. Zusätzlich steht beim FZD ein Serienradarsensor eines Automobilzulieferers zur Verfügung, welcher allerdings keinen Einblick in die genaue Funktionsweise des Trackingmoduls erlaubt. Inhalt dieser Masterthesis ist es daher eine Datenverarbeitungskette zu implementieren, die sowohl aus realen als auch aus synthetischen Daten Objekte extrahiert und als Objektliste ausgibt.

⁸ Wachenfeld, W.; Winner, H.: Die Freigabe des autonomen Fahrens (2015), a: S. 458; b: S. 461f.

⁹ Der Referenzwert zur Berechnung der Teststrecke, welcher das Unfallrisiko beschreibt, sinkt mit zunehmender Verkehrssicherheit, sodass die zu fahrenden Testkilometer steigen.

¹⁰ Winner, H.; Wolf, G.: Quo Vadis, FAS? (2015), S. 669.

1.2 Konkretisierung der Aufgabenstellung

Im Rahmen dieser Masterthesis ist ein Algorithmus zur Radar-Objekterkennung und Tracking zu implementieren, welcher in der Lage ist sowohl Mess- als auch Simulationsdaten derart zu verarbeiten, dass eine Objektliste ausgegeben wird. Vor der Implementierung ist eine umfassende Literaturrecherche über den Stand der Technik von Radar-Objekterkennungs- und Trackingverfahren durchzuführen. Die aus der Übersicht gewonnenen Methoden für die einzelnen Schritte des Trackings werden bezüglich der Anwendung innerhalb dieser Arbeit bewertet und ausgewählt. Zusätzlich ist eine Einarbeitung in die Funktionsweise des Radarsensormodells sowie der Simulationsumgebung obligatorisch. Der implementierte Tracking-Algorithmus muss dabei in der Lage sein, auftretende Ambiguitäten des verwendeten Radarsensormodells und des realen Radarsensors aufzulösen. Zur Verifikation des implementierten Verfahrens sind Testszenarien zu definieren, welche zum einen die Leistung des implementierten Algorithmus identifizieren und zum anderen die Übertragbarkeit von Mess- und Simulationsergebnissen durch deren Vergleich ermöglicht. Als Referenz zur Leistungsverbesserung gilt der am FZD vorhandene Radarsensor.

1.3 Methodik des Vorgehens

Das Vorgehen zur Bearbeitung ist in folgender Abbildung dargestellt und beginnt, wie bereits in Kapitel 1.2 erwähnt, mit einer Einarbeitung in die Thematik der Objekterkennung und des Trackings. Diese Einarbeitung umfasst eine Literaturrecherche zum Stand der Technik von Objekterkennungs- und Trackingverfahren als auch eine Einarbeitung in das vorhandene Sensormodell und dessen Simulationsumgebung in Vires VTD.

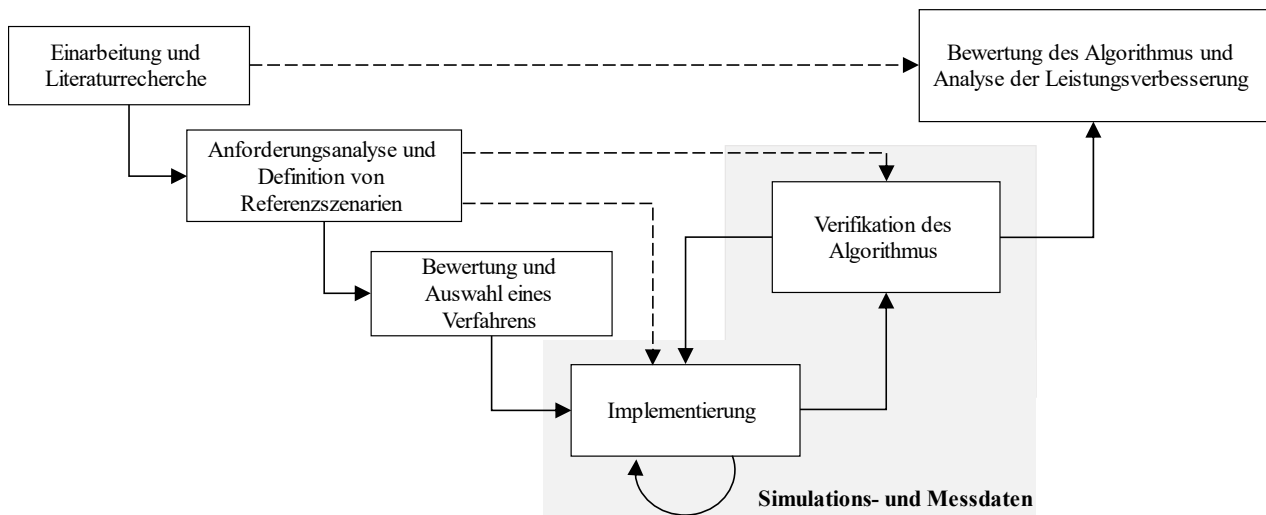


Abbildung 1-1: Darstellung der Methodik des Vorgehens

Eine anschließende Anforderungsanalyse an den zu implementierenden Algorithmus und die Definition von Referenzszenarien geben die Grundlage zur Bewertung der Verfahren aus der gewonnenen Übersicht und einer anschließenden Auswahl zur Implementierung. Zur Bewertung werden in den Anforderungen unter anderem die Eingabe- und Ausgabedaten des Algorithmus definiert als auch

Kriterien bezüglich der Komplexität, der Implementierbarkeit innerhalb dieser Arbeit und der Leistungsverbesserung im Vergleich zum vorhandenen Radarsensor festgelegt.

Das ausgewählte Verfahren wird nachfolgend implementiert und mit ersten Simulations- und Messdaten iterativ optimiert und abschließend verifiziert. Neben den Simulationsdaten stehen dazu Messdaten bereits durchgeführter realer Testfahrten in einer Datenbank am Fachgebiet zur Verfügung. Bei Bedarf werden weitere Messdaten aufgenommen.

Um den Algorithmus nach der Verifikation zu bewerten, werden die zuvor definierten Referenzszenarien verwendet, welche die Grenzen des Verfahrens hervorheben und zusätzlich eine Analyse der Leistungsverbesserung zu dem vorhandenen Radarsensor erlauben. Während des gesamten Implementierungs- und Verifikationsprozesses sowie für eine abschließende Bewertung des implementierten Verfahrens erfolgt laufend ein Abgleich mit den zuvor definierten Anforderungen.

2 Grundlagen

Um eine Grundlage für das Verständnis von Tracking-Algorithmen zu schaffen, welche auf Sensordaten basieren, wird nachfolgend auf die Radarsensorik (2.1) eingegangen. Neben den Aufgaben (2.1.1), des Aufbaus (2.1.2) und der Messprinzipien für die einzelnen Messgrößen (2.1.3) werden auch die für diese Arbeit relevante Schritte in der Signalverarbeitungskette (2.1.4) und die daraus resultierenden Mehrdeutigkeiten der Messgrößen (2.1.5) erläutert. Das Kapitel schließt mit einer Beschreibung des am FZD entwickelten Radarsensormodells (2.2) und der darin verwendeten Methoden ab.

2.1 Radarsensorik

Der Radarsensor, dessen Bezeichnung eine Kurzform für „Radio Detection and Ranging“ ist, wurde ursprünglich zu Militärzwecken entwickelt und während des zweiten Weltkrieges auf dem Gebiet der Ortungstechnik eingesetzt. Die anschließende kontinuierliche Weiterentwicklung ermöglichte den außermilitärischen Einsatz unter anderem in der Luft- und Seefahrt als auch im Straßen- und Schienenverkehr. Nach jahrzehntelanger Forschung konnte so im Jahr 1998 erstmals ein Fahrzeug mit einer radarbasierten adaptiven Geschwindigkeitsregelung (engl.: Adaptive Cruise Control, ACC) auf den Markt gebracht werden. Wenige Jahre später folgte die Entwicklung der automatischen Notbremse und der Fahrstreifenwechselassistenten.¹¹

2.1.1 Aufgaben

Zu den Aufgaben des Radarsensors gehört primär die Entdeckung von Objekten, welche sich in ihren elektromagnetischen Eigenschaften von denen des Ausbreitungsmediums unterscheiden. Aus der Entdeckung eines Objekts folgt die Bestimmung der Objektparameter, also der Entfernung, der Richtung und der Relativgeschwindigkeit. Weiterhin kann aus der Rückstrahlintensität auf die Größe des Objektes oder dessen Beschaffenheit geschlossen werden. Als weitere Aufgabe des Radarsensors ist die Zielverfolgung zu nennen, bei der die Position eines einzelnen Objektes fortlaufend überwacht wird.¹² Diese zeitliche Nachverfolgung nennt sich Tracking und stellt den Kern dieser Arbeit dar, sodass im weiteren Verlauf der Arbeit näher auf diese Thematik eingegangen wird.

¹¹ Winner, H.: Radarsensorik (2015), S. 260.

¹² Detlefsen, J.: Radartechnik (1989), S. 1.

2.1.2 Aufbau

Der am häufigsten verwendete Radarsensortyp ist ein Pulsradar, welcher in Abbildung 2-1 dargestellt ist.

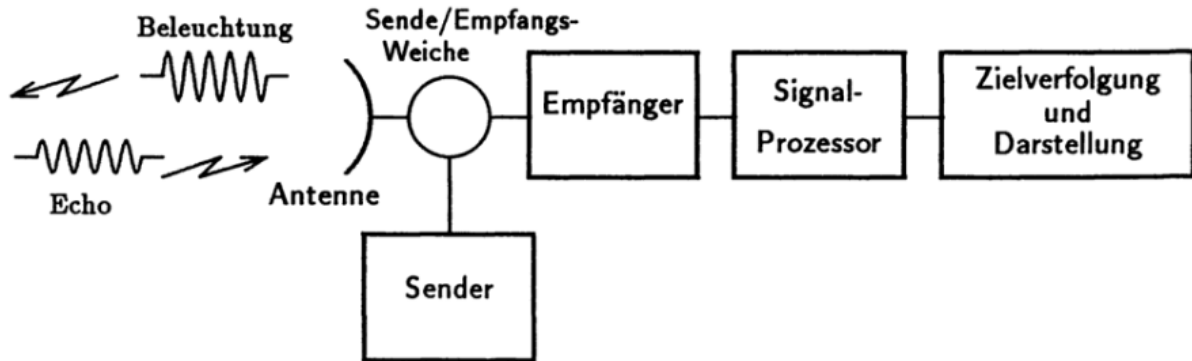


Abbildung 2-1: Prinzip des Pulsradars¹³

Ein Sender erzeugt hierbei Trägerpulse (etwa 1 μ s) elektromagnetischer Wellen, welche über die Sende-Empfangs-Weiche auf die Antenne gelangen und von ihr mit hoher Richtwirkung abgestrahlt werden (Beleuchtung). Treffen diese Wellen auf ein Objekt, wird ein Teil der Energie zum Radar reflektiert (Echo) und bildet das Streufeld, das von der Antenne empfangen wird. Von dort gelangt das empfangene Echo über die Sende-Empfangs-Weiche zum Empfänger, welcher das Empfangssignal rauscharm verstärkt und dem Signalprozessor zur Ermittlung der Objektparameter und Unterdrückung von Störsignalen, dem sogenannten Clutter, zuleitet.¹³

Die Richtwirkung der Strahlung wird durch eine antennenabhängige Bündelung erreicht, d.h. die Wellen, die den Sensor verlassen, weisen nicht die gleiche Intensität in alle Raumrichtungen auf, wie es bei einer isotropen Kugelwelle der Fall wäre. Die Kenngröße der Bündelung ist der sogenannte direkte Antennengewinn G_D , welcher proportional zur Bündelung der Strahlen ist. Er beschreibt das Verhältnis zwischen der Intensität $P(\alpha, \vartheta)_{max}$ im Raumwinkel der stärksten Abstrahlung und dem Wert $\frac{P_{total}}{4\pi}$ eines homogenen Kugelstrahlers gleicher Gesamtleistung¹⁴

$$P_{total} = \iint P(\alpha, \vartheta) d\phi d\vartheta. \quad (2-1)$$

Hierbei ist α der Azimutwinkel in der horizontalen Ebene, während ϑ der Elevationswinkel in der vertikalen Ebene ist.

2.1.3 Messgrößen und Messprinzipien

Zur Ermittlung der Objektparameter werden unterschiedliche Informationen aus dem empfangenen Echo extrahiert. Die ausgesendete und empfangene Welle ist dabei nur Träger der Information. Die Information selbst, welche z.B. für die Messung der Entfernung benötigt wird, muss der Trägerwelle

¹³ Ludloff, A.: Handbuch Radar (1993), S. 1–2.

¹⁴ Winner, H.: Radarsensorik (2015), S. 260.

sendeseitig aufmoduliert und empfangsseitig demoduliert werden. Diese Modulation und Demodulation ermöglichen die Wiedererkennung einer empfangenen Welle und den daraus resultierenden Nutzen eines Zeitbezugs zur Messung der Laufzeit. Zur Modulation der Welle stehen die drei Variablen der harmonischen Wellenfunktion Amplitude A , Frequenz f_0 und Phase ϕ zur Verfügung. Radarsysteme im Automobil nutzen Amplituden- sowie Frequenzmodulationen. Das aufmodulierte Signal kann dabei beispielsweise ein Puls oder Sinussignal sein, welches in der folgenden Abbildung für beide Modulationsarten dargestellt ist.^{15a} Im weiteren Verlauf wird auf die Ermittlung der einzelnen Messgrößen eines Radarsensors eingegangen.

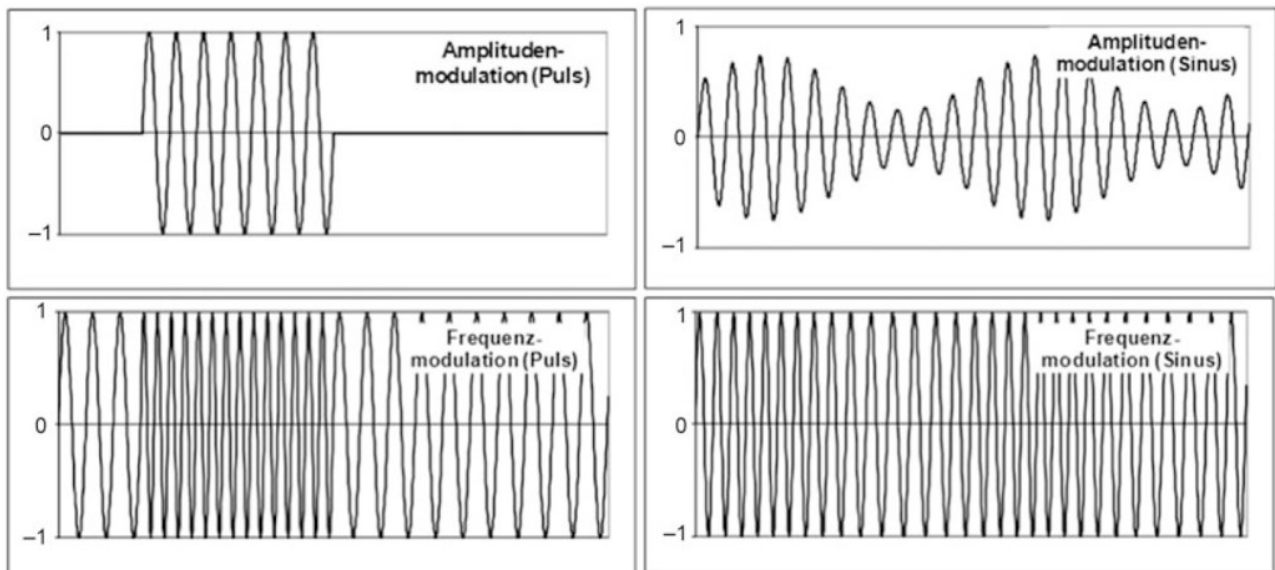


Abbildung 2-2: Idealisierte Darstellung einer Amplituden- und Frequenzmodulation.; links: Aufmodulation eines Pulssignals; rechts: Aufmodulation eines Sinussignals^{15a}

2.1.3.1 Entfernung

Der am Fachgebiet vorliegende Radarsensor basiert auf dem Pulskompressionsverfahren. Daher wird die Entfernungsmessung dieses Radarsystem erläutert. Beim Pulskompressionsverfahren, welches auch Chirp-Sequence-Modulation heißt, wird die Frequenz des gesendeten Signals in aufeinanderfolgenden Rampen kontinuierlich variiert.^{15b} Die Frequenzrampen eines Send- und Empfangssignals sind in Abbildung 2-3 beispielhaft dargestellt. Darin ist neben der Rampensteigung m auch die Differenzfrequenz Δf zwischen Empfangs- und Sendesignal zu erkennen. Diese Differenzfrequenz liegt bei

$$\Delta f = m \cdot \Delta t \quad (2-2)$$

wobei Δt die vergangene Zeit zwischen dem Senden und Empfangen des Signals, der sogenannten „time of flight“, ist. Diese Größe lässt sich mit der Entfernung zum Ziel r und der Lichtgeschwindigkeit c auch wie folgt ausdrücken:

¹⁵ Winner, H.: Radarsensorik (2015), a: S. 264; b: S. 270.

$$\Delta t = \frac{2r}{c} \quad (2-3)$$

Setzt man nun die Gleichung (2-2) in (2-3) ein und löst nach r auf, ergibt sich für die Entfernung zum Ziel

$$r = \frac{\Delta f}{2m} c \quad (2-4)$$

Mit der gemessenen Differenzfrequenz Δf und der bekannten Rampensteigen m lässt sich so die Entfernung zum Ziel berechnen.¹⁶

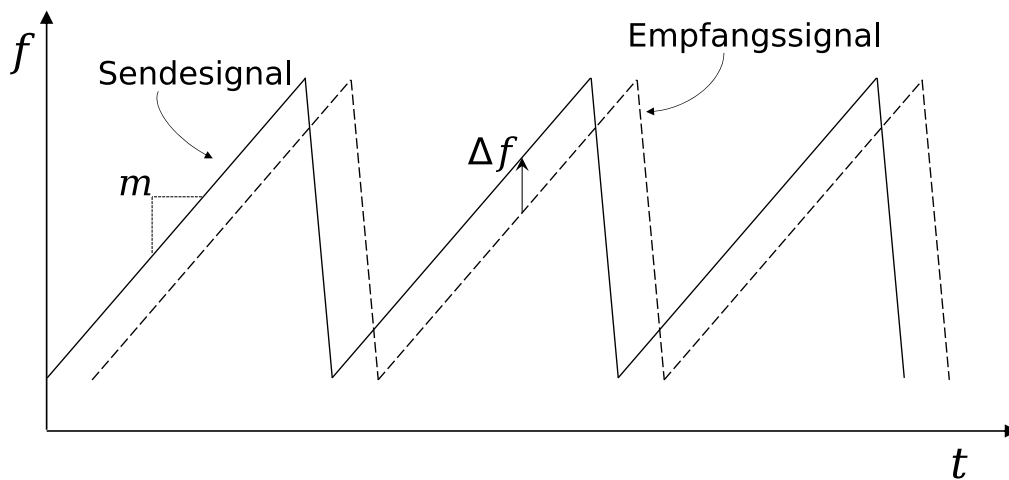


Abbildung 2-3: Frequenzrampen eines Pulskompressionsradars¹⁷

2.1.3.2 Radiale Relativgeschwindigkeit

Die radiale Relativgeschwindigkeit wird über den Dopplereffekt berechnet, da die ausgesendete elektromagnetische Welle eine Frequenzverschiebung erfährt, sobald es auf ein Objekt trifft, das sich relativ zum Radarsensor bewegt. Diese Frequenzverschiebung ergibt sich durch die Änderung der Phasenverzögerung infolge einer wachsenden oder sinkenden Entfernung zum Objekt. Die Frequenzverschiebung f_{dop} ist proportional zur radialen Relativgeschwindigkeit \dot{r} und zum Kehrwert der Wellenlänge λ und lässt sich ausdrücken durch:

$$f_{\text{dop}} = -\frac{2\dot{r}}{\lambda} \quad (2-5)$$

Die Frequenz der empfangenen Welle ergibt sich damit aus der Trägerfrequenz f_0 der gesendeten Welle und der Dopplerfrequenz zu¹⁸

¹⁶ Andrew Stove, G.: Obstacle detection radar for cars (1991), S. 237.

¹⁷ nach Winner, H.: Radarsensorik (2015), S. 277, Abbildung 17.17.

¹⁸ Winner, H.: Radarsensorik (2015), S. 264.

$$f = f_0 + f_{\text{dop}} = f_0 - \frac{2\dot{r}}{\lambda} \quad (2-6)$$

Obwohl für die einzelnen Rampen des Pulskompressionsradars eine Frequenzverschiebung aus der Dopplerfrequenz erfolgt, ist diese aufgrund der kurzen Dauer der Rampen zur Entfernungsmessung vernachlässigbar. Es gibt damit innerhalb einer Rampendauer eine eindeutige Zuordnung von Entfernung zur Differenzfrequenz. Die Dopplerfrequenz wird aus den komplexen Amplituden aufeinanderfolgender Rampen bestimmt.^{19a}

2.1.3.3 Richtung

Die Richtung des erfassten Objekts in der horizontalen Ebene wird durch den Azimutwinkel angegeben. Um diesen Winkel zu bestimmen, stehen unterschiedliche Verfahren zur Verfügung.

Eine Methode stellt das Scanning dar. Hierbei wird die Antenne innerhalb eines Mess- und Auswertezyklus mechanisch über den gesamten azimutalen Erfassungsbereich geschwenkt. Zwar weist diese Methode aufgrund ihrer hohen Genauigkeit einen Vorteil gegenüber anderen Verfahren auf, allerdings verschlechtert sich die Trennschärfe für die Dopplerauswertung aufgrund der kurzen Messdauer der einzelnen Winkelsegmente, sodass die Relativgeschwindigkeitsmessung ungenauer ist als die einer Mehrstrahlanordnung bei gleicher Messzeit. Einen weiteren Nachteil stellt die mechanische Strahlableitbarkeit selbst aufgrund ihrer verschleißenden Bewegung dar.^{19b}

Das Monopulsverfahren hingegen besteht aus einer starren Antennenanordnung. Dabei werden zwei Empfangsantennen so positioniert, dass sie horizontal um einen Faktor Γ der Wellenlänge λ verschoben sind. Dadurch trifft das reflektierte Signal nicht gleichzeitig, sondern zeitlich versetzt auf die Empfangsantennen. Dieser Versatz resultiert in einer Phasenverschiebung $\Delta\phi$ in den Empfangssignalen beider Antennen, welcher abhängig vom Azimutwinkel α ist. Durch Umformen der Beziehung

$$\Delta\phi = 2\pi\Gamma\lambda \cdot \sin(\alpha) \quad (2-7)$$

wird der Azimutwinkel ermittelt. Um die Eindeutigkeit einer Phasenverschiebung zu gewährleisten, muss allerdings der maximale Winkelmessbereich in Abhängigkeit des Antennenabstands $\Gamma\lambda$ auf $2\pi\Gamma\lambda \cdot \sin(\alpha_{\text{max}}) < \frac{\pi}{2}$ beschränkt werden, wobei α_{max} den maximalen Azimutwinkel darstellt.^{19c}

Um diese Einschränkung des Winkelmessbereichs im Monopulsverfahren zu verbessern, werden sogenannte Mehrstrahler genutzt. Dabei wird durch den Einsatz von Antennen unterschiedlicher Antennencharakteristiken verschiedene Winkelbereiche abgedeckt. Planar-Antennen-Arrays sind eine Art von Mehrstrahlern. Diese Variante nutzt statt Stabantennen eine Anordnung von mehreren Sende- und Empfangsantennenflächen, den sogenannten Patches, welche die Bautiefe des Sensors erheblich verkleinern. Durch die kontinuierliche Ansteuerung der Antennenelemente bei angepasster Phasendifferenz lässt sich ein elektronisch schwenkbares Scanning ermöglichen, bei dem die Nachteile eines mechanischen Scannings wegfallen.^{19d} Der Radarsensor des Automobilzulieferers, welcher am Fachgebiet vorliegt, basiert auf dieser Methode der Planar-Antennen-Arrays.

¹⁹ Winner, H.: Radarsensorik (2015), a: S. 277–278; b: S. 280–281; c: S. 281–282; d: S. 283–284.

2.1.3.4 Rückstrahlintensität

Die Rückstrahlintensität eines erfassten Objekts gibt die Leistung des Empfangssignals an, welche nach der Reflexion am Ziel vom Sensor aufgenommen wird. Die aufgenommene Leistung hängt dabei von mehreren Faktoren ab, welche nachfolgend zur Radargleichung zusammengefasst werden. Zum einen unterliegen elektromagnetische Wellen einer atmosphärischen, witterungsabhängigen Dämpfung. Der Verlust durch atmosphärische Dämpfung ist proportional zur Entfernung r zum Objekt und wirkt mit dem Faktor $10^{\frac{-2kr}{1000}}$ auf die Empfangsleistung. Der Faktor k ist die witterungsabhängige Absorption, welche allerdings in vielen Fällen so gering ist, dass sie überwiegend vernachlässigt wird.^{20a}

Die Leistungsdichte der vom Sensor ausgesendeten Strahlung an einem Punkt mit dem Abstand r lässt sich in Abhängigkeit des Antennengewinns G_S in Richtung der Strahlung und der Gesamtleistung P_{total} des Sensors mit der folgenden Beziehung angeben:²¹

$$\frac{P_{\text{total}} \cdot G_S}{4\pi \cdot r^2} \quad (2-8)$$

Die Strahlung mit dieser Leistungsdichte wird nun von einem Objekt mit dem Radarquerschnitt σ aufgenommen und in alle Raumrichtungen reflektiert. Der Radarquerschnitt (engl.: Radar Cross Section, RCS) eines Objekts entspricht der Querschnittsfläche eines hypothetischen, isotropen Kugelreflektors, der in einer Entfernung r den gleichen Anteil der Strahlung zum Radar zurückreflektiert wie das erfasste Objekt.²² Damit wird die Leistungsdichte des Echos am Radar zu²³

$$\frac{P_{\text{total}} \cdot G_S}{4\pi \cdot r^2} \cdot \frac{\sigma}{4\pi \cdot r^2} \quad (2-9)$$

An der Empfangsantenne wirkt nun, analog zum Antennengewinn der Sendeantenne, eine Antennenverstärkung G_E auf die aufgenommene Strahlung.²⁴

Der Einfluss von Interferenz aufgrund von Mehrwegeausbreitung und der damit verbundenen Änderungen der Signalleistung durch konstruktive oder destruktive Interferenz wird mit einem Schüttelfaktor V_{mp}^2 , wobei $0 \leq V_{\text{mp}} \leq 2$, berücksichtigt.

Die Radargleichung für die Empfangsleistung P_E ergibt sich damit aus den genannten Faktoren zu^{20b}

$$P_E = \frac{10^{\frac{-2kr}{1000}} \cdot \sigma \lambda^2 \cdot G_S \cdot G_E \cdot V_{\text{mp}}^2 \cdot P_{\text{total}}}{(4\pi)^3 \cdot r^4} \quad (2-10)$$

²⁰ Winner, H.: Radarsensorik (2015), a: S. 262; b: S. 263.

²¹ Skolnik, M. I.: Introduction to radar systems (1980), S. 3.

²² Rezende, M. C. et al.: Radar Cross Section Measurements (2002), S. 24.

²³ Ludloff, A.: Handbuch Radar (1993), S. 2–3.

²⁴ Richards, M. A.: Radar signal processing (2005), S. 13.

2.1.4 Signalverarbeitung

Die einzelnen Schritte der Signalverarbeitung im Radarsensor erfolgen für verschiedene Modulations- und Antennenkonzepte stets in der gleichen Abfolge, welche in Abbildung 2-4 dargestellt ist. Begonnen wird dabei bei der Strahlformung und Modulation der auszusendeten Signale. Empfangene Signale unterlaufen anschließend einer Vorverarbeitung und Digitalisierung zur Datenerfassung und Identifikation der Phasenverschiebungen. Um aus den erfassten Phasenverschiebungen die einzelnen Messgrößen zu ermitteln, wird eine Spektralanalyse per Fouriertransformation durchgeführt. Auf dieser Grundlage erfolgt die Extraktion der Detektionen aus allen Empfangssignalen. Rauschen und Signalstörungen werden so weitestgehend gefiltert. Die weitere Verarbeitung der Detektionen innerhalb eines Radarsensors ist nicht festgelegt und kann auf unterschiedliche Weise erfolgen. Möglich ist hier die Verarbeitung in Form einer Bündelung, dem Zusammenfassen von Detektionen, die zu einem Objekt gehören, und des Trackings, der zeitlichen Nachverfolgung von Objektdetektionen.²⁵ Nachfolgend wird auf die Schritte der Spektralanalyse und der Detektionsextraktion näher eingegangen, da der Tracking-Algorithmus an diesem Punkt der Signalverarbeitungskette ansetzt.

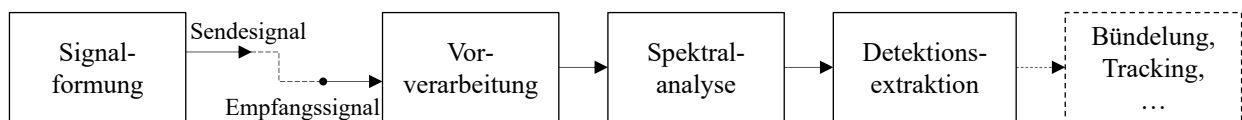


Abbildung 2-4: Signalverarbeitungskette eines Radarsensors

2.1.4.1 Spektralanalyse

Um die Frequenzverschiebungen des Empfangssignals zur Bestimmung der Objektparameter zu identifizieren, wird das Empfangssignal aus dem Zeitbereich mithilfe einer Fouriertransformation in den Frequenzbereich überführt. Da das Empfangssignal nicht als kontinuierlicher Verlauf vorliegt, sondern mit einer diskreten Abtastrate erfasst wird, wird eine schnelle Fouriertransformation (engl.: Fast Fourier Transformation, FFT) auf das Signal im Zeitbereich angewendet. In Abbildung 2-5 ist das Ergebnis einer schnellen Fouriertransformation beispielhaft dargestellt. Nach der Transformation liegen die Frequenzanteile des Empfangssignals $h(t)$ als Spektrallinien in $F\{h(t)\}$ vor, welche aufgrund deren Periodizität im positiven und negativen Bereich auftreten. Deren Betrag entspricht dabei der Amplitude des jeweiligen Frequenzanteils.²⁶

²⁵ Winner, H.: Radarsensorik (2015), S. 291.

²⁶ Detlefsen, J.: Radartechnik (1989), 121f.

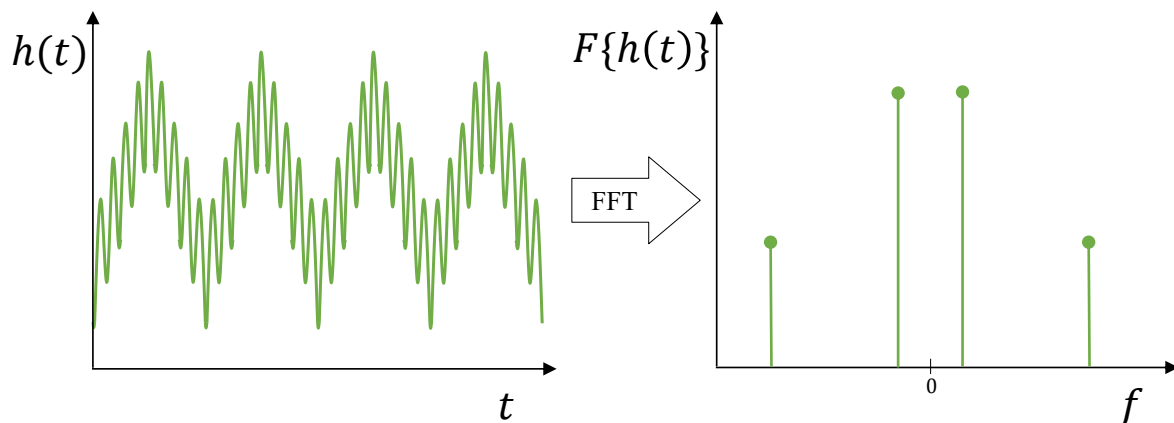


Abbildung 2-5: Überführung des Empfangssignals aus dem Zeitbereich (links) in den Frequenzbereich (rechts)

Damit beim verwendeten Pulskompressionsradar die drei Messgrößen, Entfernung, Relativgeschwindigkeit und Azimutwinkel, bestimmt werden können, wird eine mehrdimensionale FFT durchgeführt, bei der drei FFT aufeinander folgen. Die erste FFT identifiziert hierbei die Differenzfrequenz zwischen den gesendeten und empfangenen Frequenzrampen. Mit der eindeutigen Zuordnung der Differenzfrequenz einer Rampe zu einer Entfernung des reflektierenden Objekts wird das Ergebnis als entsprechender Entfernungsbins dargestellt, die in Abbildung 2-6 als graue Zellen dargestellt sind.^{27a} Die Größe eines Bins bzw. Zelle bestimmt die Trennfähigkeit des Sensors für den jeweiligen Objektparameter.^{27b}

Mit den Werten der ersten FFT bestimmt die Anwendung einer zweiten FFT die Dopplerfrequenz der gesamten Rampenfolge. Das Ergebnis dieser zwei FFT wird als zweidimensionales Datenfeld verstanden, bei dem die Messdaten einzelner Rampen (Differenzfrequenz) die Spalten und die einer Rampenfolge (Dopplerfrequenz) die Zeilen bilden.^{27a}

Die dritte FFT bestimmt aus der ermittelten Phasenverschiebung an den Empfangsantennen den Azimutwinkel des Empfangssignals und stellt diesen ebenfalls als Bin dar. Fasst man nun die Darstellungen der Ergebnisse aller FFT zusammen, lassen sich diese als ein Zellvolumen in einem dreidimensionalen Würfel, dem Radarwürfel, visualisieren.^{27b}

²⁷ Winner, H.: Radarsensorik (2015), a: S. 278; b: S. 288.

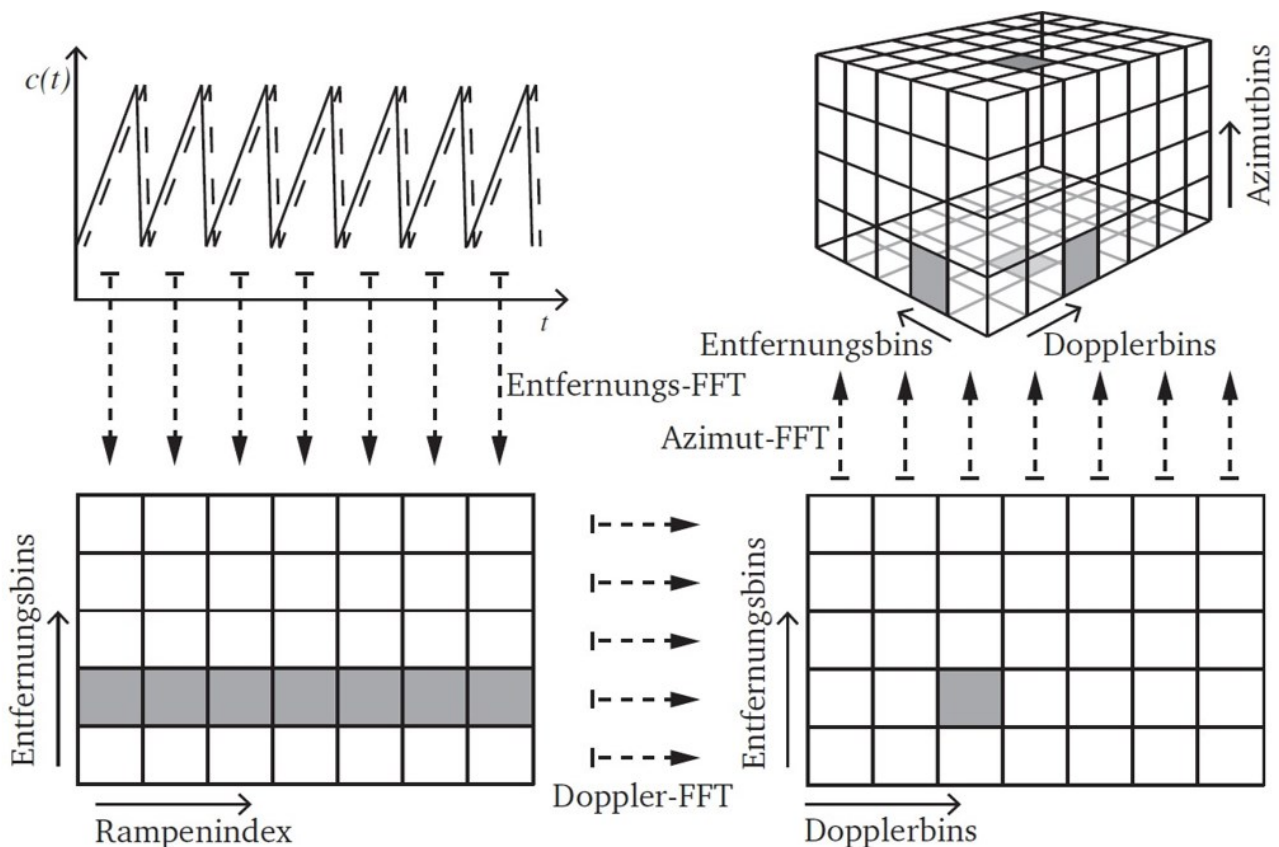


Abbildung 2-6: Dreidimensionaler FFT mit Bindarstellung²⁸

2.1.4.2 Detektion

Die im Radarwürfel enthaltenen Informationen ergeben sich nicht nur aus Objekten, die tatsächlich im Sichtfeld des Sensors vorhanden sind, sondern zusätzlich aus Rauschen oder Störsignalen. Um aus dieser Menge an Informationen, die der Radarwürfel liefert, relevante Reflexionen zu extrahieren, werden in den gemessenen Datenreihen nach besonderen Merkmalen gesucht. Dies erfolgt durch den Vergleich der Leistungswerte der Reflexionen mit einem definierten Schwellenwert. Liegt der Leistungswert einer Reflexion unter dem definierten Schwellenwert, wird die Reflexion auf dieser Verarbeitungsebene gefiltert, während eine Reflexion mit einem höheren Leistungswert als Detektion klassifiziert wird.

Dieser Schwellenwert ergibt sich aus einem Schwellenalgorithmus, der möglichst alle Leistungswerte, die von realen Objekten stammen, findet, aber unempfindlich gegenüber Leistungswerten, die durch Rauschen oder Signalstörungen entstanden sind, ist. Die Schwierigkeit in diesem Schritt der Signalverarbeitung liegt in den stark unterschiedlichen Leistungswerten unterschiedlicher Objekte bzw. desselben Objekts zu verschiedenen Zeiten je nach Entfernung und Orientierung.^{29a} Eine Methode, welche hier Anwendung findet, ist die Verwendung der sogenannten Constant False-Alarm

²⁸ Linnhoff, C.: Masterthesis, Entwicklung eines Radar-Sensormodells (2018), S. 16.

²⁹ Winner, H.: Radarsensorik (2015), a: S. 292; b: S. 278.

Rate (CFAR). Hierbei wird für jeden Zyklus ein adaptiver Schwellenwert ermittelt, welcher das aktuelle Rauschen und den Einfluss von Störsignalen berücksichtigt.³⁰

2.1.5 Mehrdeutigkeiten

Die Auflösung der Entfernung- und Dopplerbins des Radarwürfels und damit der maximale Wertebereich dieser Messgrößen werden durch die Abtastfrequenz f_s und der Chirpfolgefrequenz $\frac{n_R}{T_M}$ bestimmt. Die Chirpfolgefrequenz ergibt sich aus der Anzahl der Rampen n_R eines Zyklus und der Messzeit T_M . Die maximale Entfernung ist somit durch die Beziehung

$$r_{\max} = \frac{\pi c}{4m} f_s \quad (2-11)$$

festgelegt, wobei c die Lichtgeschwindigkeit darstellt und m die Steigung einer Frequenzrampe des ausgesendeten Signals ist.

Die maximal messbare Relativgeschwindigkeit ergibt sich aus der Chirpfolgefrequenz und der Trägerfrequenz f_0 zu^{29b}

$$|\dot{r}|_{\max} = \frac{n_R}{T_M} \cdot \frac{c}{4f_0} \quad (2-12)$$

Besteht die Rampenfolge aus $n_R = 512$ Rampen bei einer Messzeit von $T_M = 20$ ms liegt die maximal messbare Relativgeschwindigkeit bei einer Trägerfrequenz von $f_0 = 77$ GHz bei $|\dot{r}|_{\max} = 24,92 \frac{\text{m}}{\text{s}}$.³¹ Da die auftretenden Relativgeschwindigkeiten im Straßenverkehr diesen Wertebereich überschreiten und das Vergrößern der Rampenzahl mit einem kleineren Entfernungsmessbereich verbunden ist, wird der Wertebereich auf Kosten von Mehrdeutigkeiten künstlich vergrößert.³² Diese Mehrdeutigkeiten entstehen bei Relativgeschwindigkeiten, deren zugehörige Dopplerfrequenz größer ist als die halbe Chirpfolgefrequenz und somit der entsprechende Dopplerbin im einfachen Wertebereich der FFT nicht direkt abgebildet werden kann.³³ Dieser erweiterte Wertebereich wird am Rand des eindeutigen Wertebereichs fortgeführt und überlappt den eindeutigen Wertebereich. Folglich besteht ein FFT mit Relativgeschwindigkeiten, die den eindeutigen Messbereich übersteigen, aus mehreren übereinanderliegenden Wertebereichen, wie in Abbildung 2-7 anschaulich dargestellt ist. Dies hat zur Folge, dass für einen Dopplerbin die Zugehörigkeit zum Wertebereich nicht eindeutig erkennbar ist. Beispielsweise wird bei einem errechneten Dopplerbin von 550 bei nur 512 vorhandenen Bins im Eindeutigkeitsbereich ein Dopplerbin von $(550 - 512) = 38$ ausgegeben. Für die Detektion mit diesem Dopplerbin kommen für einen Zyklus damit zwei unterschiedliche Relativgeschwindigkeiten in Frage, da nach der FFT nur der Wert 38 bekannt ist.

³⁰ Rohling, H.: Radar CFAR Thresholding in Clutter and Multiple Target Situations (1983), S. 608.

³¹ Die Zahlenwerte entsprechen den technischen Eigenschaften des am Fachgebiet genutzten Radarsensor.

³² Pirttilä, J. et al.: A Proposed Solution to the Range-Doppler Dilemma (2005), S. 1375.

³³ M. Kronauge et al.: Radar target detection and Doppler ambiguity resolution (2010), S. 1.

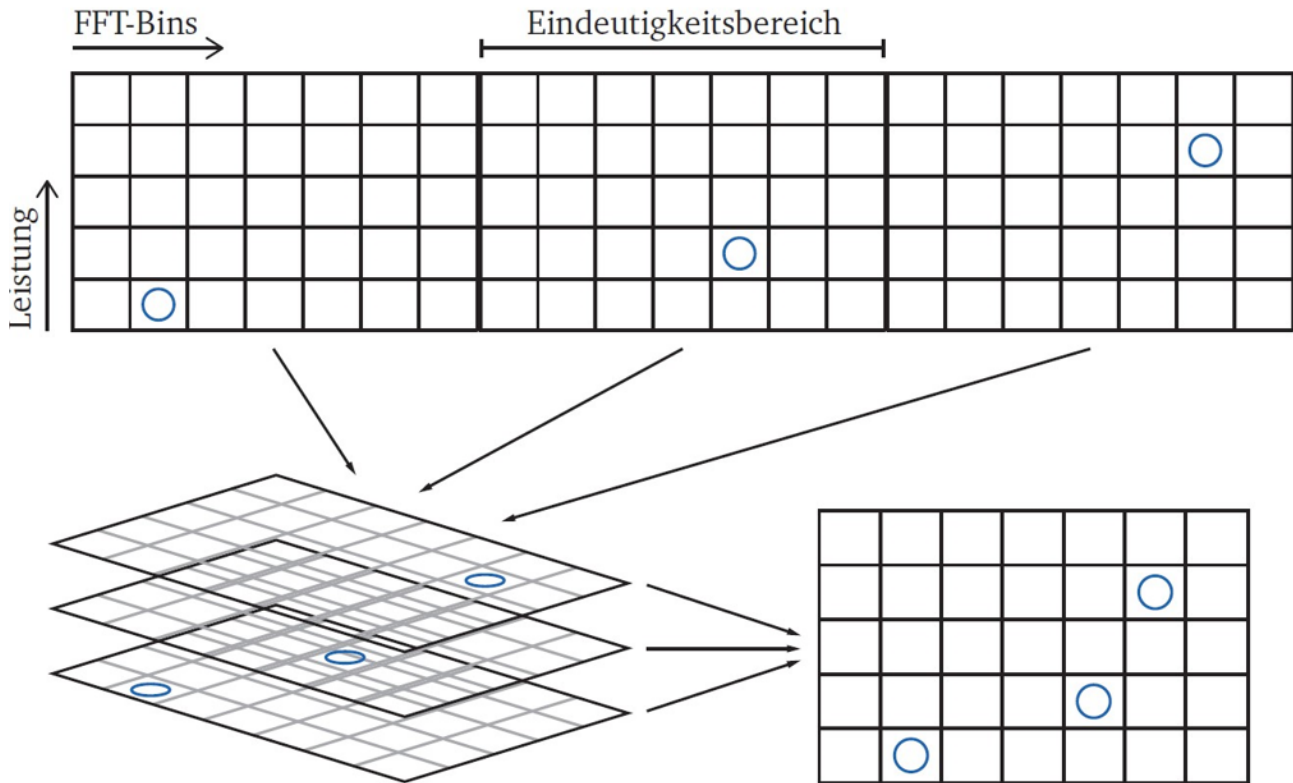


Abbildung 2-7: Erweitern des eindeutigen FFT Wertebereichs und die damit verbundene Mehrdeutigkeit³⁴

Um diese Mehrdeutigkeit aufzulösen, wird die Messzeit T_M von Zyklus zu Zyklus leicht variiert. Daraus ergeben sich leicht unterschiedliche Auflösungen $\Delta\dot{r} = \frac{c}{2f_0 T_M}$ der einzelnen Dopplerbins für aufeinanderfolgende Zyklen.³⁵ Nach dem Chinesischen Restsatz lässt sich die Mehrdeutigkeit so mithilfe von Daten aus aufeinanderfolgenden Zyklen auflösen: Innerhalb eines Zyklus werden für einen Dopplerbin Hypothesen für unterschiedliche Relativgeschwindigkeiten aufgestellt. Diese werden mit den Hypothesen der darauffolgenden Zyklen verglichen. Bei übereinstimmenden Hypothesen von mindestens zwei konsekutiven Messungen ist die Mehrdeutigkeit aufgelöst und die zur Detektion gehörenden Relativgeschwindigkeit eindeutig bestimmt.³⁶

Der Messbereich des Azimutwinkels ist ebenfalls mehrdeutig, welche durch eine Variation der Azimutauflösung gelöst wird. Der Eindeutigkeitsbereich der Azimutwinkel ist bei dem hier verwendeten Radarsensor und Radarsensormodell allerdings so groß, dass mehrdeutige Messdaten bei relativ bewegten Objekten nur vereinzelt auftreten.

³⁴ Linnhoff, C.: Masterthesis, Entwicklung eines Radar-Sensormodells (2018), S. 19.

³⁵ Winner, H.: Radarsensorik (2015), S. 278.

³⁶ M. Kronauge et al.: Radar target detection and Doppler ambiguity resolution (2010), S. 1.

2.2 Radarmodell

Das im Folgenden beschriebene Radarmodell wurde am FZD entwickelt und wird dort aktuell für weitere Forschungsarbeiten genutzt. Die genannten Funktionsweisen beziehen sich spezifisch auf dieses Radarmodell und basieren damit maßgeblich auf der Arbeit von Linnhoff.³⁷

Das genutzte Radarmodell erzeugt synthetische Radardaten für Fahrsimulationen in der Simulationssoftware VTD. Die ausgegebenen Objektparameter entsprechen den Radardaten in der Signalverarbeitungskette nach der Fouriertransformation. Damit liegen am Ausgang des Radarmodells für Detektionen Fourierbins zur radialen Entfernung, der radialen Relativgeschwindigkeit und dem Azimutwinkel sowie die Empfangsleistung vor.

Die innerhalb dieser Arbeit genutzten synthetischen Radardaten werden ausgehend von einem Sensor mit den gleichen technischen Eigenschaften, die der am Fachgebiet genutzten Industrielösung zugrunde liegen. Damit sind die maximal messbare Entfernung auf $r_{\max} = 200$ m und der maximale Betrag des Azimutwinkels auf $|\alpha|_{\max} = 13^\circ$ beschränkt, wie in Abbildung 2-8 dargestellt ist. Der Wertebereich der detektierbaren radialen Relativgeschwindigkeiten liegt zwischen $\dot{r}_{\min} = -400 \frac{\text{km}}{\text{h}}$ und $\dot{r}_{\max} = 200 \frac{\text{km}}{\text{h}}$.

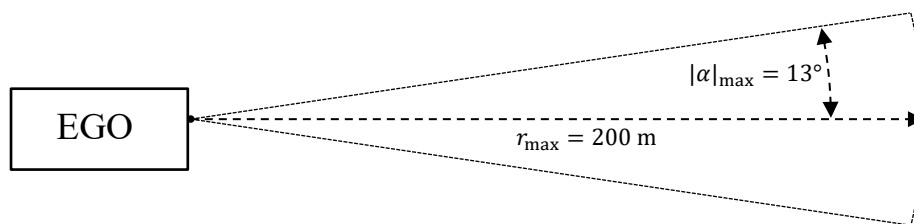


Abbildung 2-8: Sichtbereich des Radarsensors und des Radarmodells

Raytracing

Das Radarmodell basiert auf einem Raytracing-Ansatz. Das Raytracing (engl. Strahlverfolgung) ist ein Bildsyntheseverfahren, bei dem aus einer dreidimensionalen Szene ein zweidimensionales Bild erzeugt wird. Dabei ist die Blickrichtung und Position des Beobachters entscheidend, um sichtbare und verdeckte Objekte als diese identifizieren zu können. Die Grundidee des Raytracings ist die Verfolgung von Lichtstrahlen einer Lichtquelle, welche an Oberflächen reflektiert werden und so zu einer virtuellen Kamera gelangen (Forward-Raytracing). Da von einer Lichtquelle allerdings nicht alle ausgesendeten Strahlen ihren Weg zur Kamera finden, stellt dies eine sehr rechenaufwändige Methode dar. Eine Optimierung bietet das sogenannte Backward-Raytracing, welches in Abbildung 2-9 dargestellt ist. Hierbei werden von der virtuellen Kamera Strahlen ausgesendet, welche auf ihrem Weg durch den Raum zur Lichtquelle hin verfolgt werden. Jeder dieser Strahlen bildet dabei einen

³⁷ Linnhoff, C.: Masterthesis, Entwicklung eines Radar-Sensormodells (2018).

Bildpixel.³⁸ Bei der Simulation eines Radarsensors entspricht der Sensor sowohl der Lichtquelle als auch der Kamera, da ausgesendete Strahlen auch vom Sensor empfangen werden.

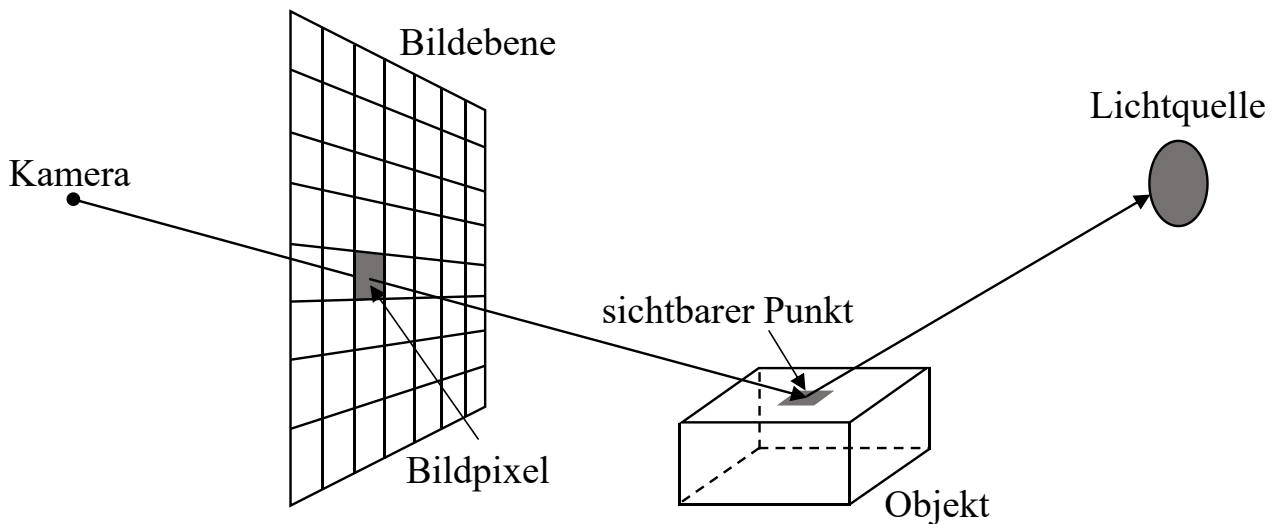


Abbildung 2-9: Backward-Raytracing³⁹

Im Radarmodell wird für das Raytracing die OptiX™ Raytracing-Engine von NVIDIA® verwendet, welche in der Simulationssoftware VTD bereits integriert ist. OptiX™ erstellt die Strahlen einzeln, welche aus einem Ursprung, einer Richtung und einer Typenbezeichnung bestehen und sendet diese anschließend aus. Trifft ein Strahl auf ein Objekt werden im sogenannten Hitprogramm die Informationen des Schnittpunkts, also des getroffenen Objekts, in Form einer Szenengraphstruktur gespeichert. Diese enthält nicht nur Informationen zu den Objekten selbst, sondern auch zu den Beziehungen der Objekte untereinander. Die räumliche Anordnung der Szene wird so festgehalten. Die gespeicherten Informationen stammen zum einen aus dem Raytracer, wie die Strahllänge und dem Empfangswinkel, und zum anderen aus der Simulationsumgebung VTD, woraus die Objektgeschwindigkeit und die Reflexionseigenschaften extrahiert werden.

Durch globale Puffer haben andere Programmteile Zugriff auf die Informationen der Szenengraphstruktur und können diese nutzen. Diese Programmteile erfassen die gespeicherten Szeneninformationen und verarbeiten sie zur finalen Form der Ausgabedaten des Radarmodells weiter.

³⁸ Glassner, A. S.: An Introduction to Ray tracing (1989), S. 5-9.

³⁹ Linnhoff, C.: Masterthesis, Entwicklung eines Radar-Sensormodells (2018), S. 21.

3 Anforderungsanalyse

Im folgenden Kapitel werden zunächst Begriffe, welche im Rahmen dieser Arbeit verwendet werden, zum einheitlichen Gebrauch abgegrenzt und definiert (3.1). Dazu gehören Begriffsdefinitionen sowie die Definition der genutzten Koordinatensysteme für den weiteren Verlauf dieser Arbeit. Ausgehend davon werden die Anforderungen (3.2) an den zu implementierenden Tracking-Algorithmus abgeleitet, welche zum späteren Zeitpunkt als Grundlage für die Identifikation geeigneter Referenzszenarien zur abschließenden Verifikation dienen.

3.1 Definition

Der folgende Abschnitt veranschaulicht das Fahrzeug- sowie Sensorkoordinatensystem. Für reale als auch synthetische Messungen stimmen diese überein.

3.1.1 Fahrzeugkoordinatensystem

Das im Rahmen dieser Arbeit genutzte Fahrzeugkoordinatensystem ist ein kartesisches Koordinatensystem und hat seinen Ursprung in der Mitte der Egofahrzeugfront, an welcher der Radarsensor montiert ist. Damit liegt die x-Achse auf Höhe der Fahrzeuglängsachse, während die y-Achse auf einer Ebene mit dem entferntesten Punkt der Fahrzeugfront liegt.⁴⁰ Blickt man als hypothetischer Fahrer in Fahrtrichtung, zeigt die x-Achse nach vorne, die y-Achse nach links und die z-Achse nach oben.

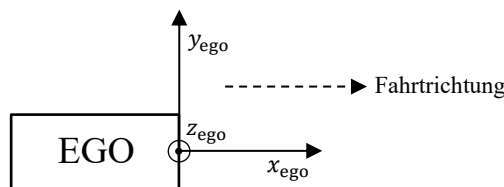


Abbildung 3-1: Fahrzeugkoordinatensystem

3.1.2 Sensorkoordinatensysteme

Beim Sensorkoordinatensystem ist zwischen dem polaren und kartesischen System zu unterscheiden. Die Ausgabedaten des Radarsensors basieren auf einem polaren Koordinatensystem, welches in Abbildung 3-2 beispielhaft dargestellt ist. Die Entfernung und ebenfalls die Relativgeschwindigkeit entsprechen damit den radialen Anteilen. Die Orientierung des kartesischen Koordinatensystems ist analog zum Fahrzeugkoordinatensystem. Aufgrund der Einbauposition des Sensors am Testfahrzeug des FZDs liegt der Ursprung beider Sensorkoordinatensysteme bei $x_{ego} = 0,18$ m.

⁴⁰ In VTD liegt der Ursprung des Fahrzeugkoordinatensystems 3,585 m vor dem Mittelpunkt der Fahrzeughinterachse.

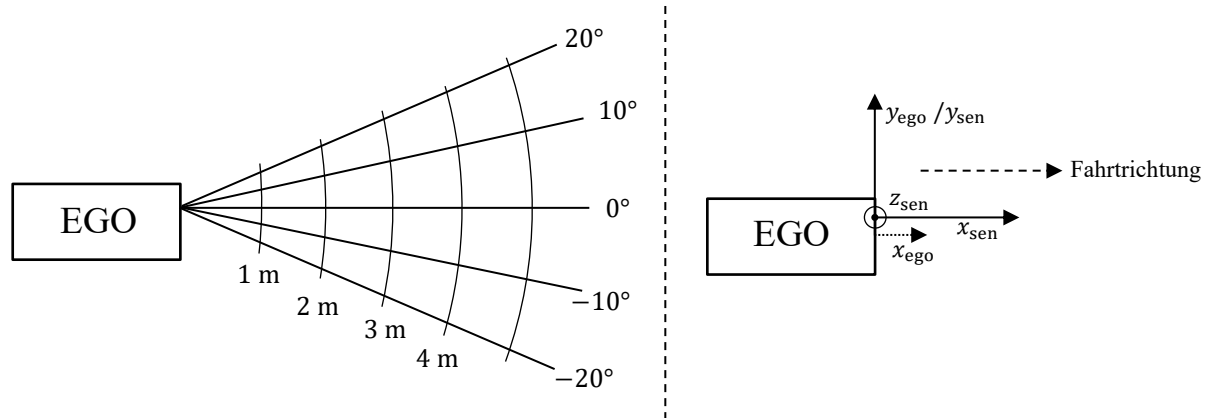


Abbildung 3-2: Polares (links) und kartesisches (rechts) Sensorkoordinatensystem

3.1.3 Objekt

Ein Objekt beschreibt innerhalb dieser Arbeit ein Element der Fahrszene, welches vom Sensor erfasst, als mindestens eine Detektion ausgegeben und anschließend vom Tracking-Algorithmus als relevant für das Egofahrzeug klassifiziert wurde. Die Relevanz beschränkt sich hierbei nicht auf das aktuelle Fahrmanöver des Egofahrzeuges, sodass beispielsweise bei einer Folgefahrt nur Fahrzeuge auf dem eigenen Fahrstreifen als Objekt erkannt werden, sondern umfasst ebenso umliegende Objekte, die nicht direkt an der Fahrsituation beteiligt sind. Ziel der Klassifikation ist damit aus den Detektionen Objekte zu extrahieren, welche eine Kollision verursachen, wenn das Egofahrzeug und dieses Objekt aufeinandertreffen. Dazu gehören statische Objekte, z.B. parkende Fahrzeuge am Fahrbahnrand, und dynamische Objekte wie Fahrzeuge auf der Fahrbahn, aber auch Fußgänger auf dem Gehweg, insofern diese im Sichtbereich des Sensors liegen. Ziel ist es, Straßenreflexionen oder Vegetation am Fahrbahnrand nicht als Objekte zu erkennen.

3.1.4 Tracking

Der folgende Abschnitt zum Tracking dient als erste Definition. Eine detaillierte Beschreibung der Thematik erfolgt im Kapitel 4. Das Tracking beschreibt die zeitliche Nachverfolgung von Einzeldetektionen, welche zu Objekten zusammengefasst werden. Dazu werden die Zustände erkannter Objekte prädictiert und anschließend mit Detektionen aus dem Sensor des darauffolgenden Messzyklus verglichen. Stimmen die prädictierten Zustände mit den Detektionen überein, werden die Detektionen dem zuvor erkannten Objekt zugewiesen.⁴¹ Alle vom Tracking-Algorithmus erkannten Objekte und ihre Zustände werden in einer Objektliste gespeichert und ausgegeben. Auf die Objektliste und ihren Inhalt wird in Kapitel 5 näher eingegangen.

⁴¹ Granström, K. et al.: Introduction to Extended Object Tracking (2016), S. 1.

3.2 Ableitung der Anforderungen

Die Anforderungen an den zu implementierenden Tracking-Algorithmus ergeben sich einerseits aus der Betrachtung üblicher Fahrszenarien, wie sie im Stadtverkehr aber auch außerorts auftreten, andererseits aus dem Verwendungszweck dieses Trackers⁴². Die Zahl der Objekte, welche sich im Sichtfeld des Sensors aufhalten, ist in solchen Szenarien weder initial bekannt noch bleibt diese zu jeder Zeit konstant. Objekte können in das Sichtfeld eintreten und es genauso verlassen. Zusätzlich können sie innerhalb des Sichtfelds von weiteren Objekten verdeckt werden. Daher ist für den Tracker festzulegen, dass dieser eine initial unbekannte Anzahl von Objekten verfolgen muss. Ausgehend davon müssen das Eintreten und Verschwinden von Objekten im Sichtfeld, durch das Generieren neuer Objekte und das Löschen alter Objekte berücksichtigt werden. Da mit dem Tracking-Algorithmus synthetische sowie reale Radarsensordaten⁴³ verwendet werden, müssen beide Datenarten derart verarbeitet werden, dass eine Objektliste ausgegeben wird. In der nachfolgenden Tabelle sind alle Anforderungen zusammenfassend gelistet. Zur Prüfung der Anforderungen werden geeignete Verifikationsmethoden definiert, welche ebenfalls in der nachfolgenden Tabelle zu finden sind.

Tabelle 1: Anforderungen an den zu implementierenden Tracking-Algorithmus

Nr.	Anforderung
1	<p>Der Tracker verarbeitet sowohl Mess- als auch Simulationsdaten und erkennt anhand von Detektionen Objekte und verfolgt diese.</p> <p>Verifikationsmethode: Radardaten des gleichen Szenarios werden als Mess- und Simulationsdaten als Eingangsgröße im Algorithmus genutzt und die Ausgangsgrößen verglichen.</p>
2	<p>Für jedes verfolgte Objekt sind anhand der Detektionen geschätzte kinematische und geometrische Eigenschaften sowie die geschätzte Objektposition bekannt und werden in der Objektliste ausgegeben.</p> <p>Verifikationsmethode: Die Ausgangsgröße des Algorithmus nach Verarbeitung der Sensordaten eines Szenarios wird mit Realdaten der Zielobjekte desselben Szenarios verglichen.</p>
3	<p>Der Tracker verfolgt eine unbekannte Anzahl von Objekten.</p> <p>Verifikationsmethode: Zwei Szenarien mit einer unterschiedlichen Zahl an Zielobjekten werden vom Tracking-Algorithmus verarbeitet.</p>
4	<p>Der Tracker erkennt Elemente, welches in das Sichtfeld des Sensors eintreten, anhand erzeugter Detektionen nach spätestens fünf Messzyklen als neue Objekte.</p> <p>Verifikationsmethode: Szenarien, in denen Zielobjekte in das Sensorsichtfeld eintreten, werden verarbeitet und die zugehörige Objektlisten untersucht.</p>
5	<p>Der Tracker löscht Objekte, welche sich nicht mehr im Sichtfeld des Sensors aufhalten, nach zehn Messzyklen.</p> <p>Verifikationsmethode: Szenarien, in denen Zielobjekte das Sensorsichtfeld verlassen, werden verarbeitet und die zugehörige Objektlisten untersucht.</p>

⁴² Tracker und Tracking-Algorithmus werden im weiteren Verlauf der Arbeit synonym verwendet.

⁴³ Synthetische und reale Radarsensordaten werden im weiteren Verlauf der Arbeit jeweils als Simulations- und Messdaten bezeichnet.

4 Tracking-Algorithmen

Tracking-Algorithmen wurden ursprünglich für Anwendungen entwickelt, welche vom Sensor weit entfernte Objekte betrachten, wie es in der Luftraumüberwachung beispielsweise der Fall ist. Mittlerweile sind Trackingverfahren eine Schlüsseltechnologie in Robotik- und Überwachungssystemen sowie in Automobilanwendungen, insbesondere im Bereich des automatisierten Fahrens.⁴⁴ Der Inhalt dieser Arbeit beschränkt sich auf Tracking-Algorithmen auf Basis von Radarsensordaten für Automobilanwendungen. Im Verlauf dieses Kapitels wird eine Einführung in das Tracking gegeben (4.1), welche einen Überblick der probabilistischen Grundlagen gibt. Anschließend folgt die Definition verschiedener

Tracking-Arten (4.2). Die allgemeine Systemarchitektur eines Tracking-Algorithmus wird dargestellt (4.3), sodass dieses Kapitel mit der Erläuterung und der Auswahl unterschiedlicher Methoden zur Implementierung des Algorithmus (4.4 - 4.7) abschließt.

4.1 Einführung in das Tracking

Unter Tracking wird die Identifikation des zeitlichen Zusammenhangs separater Sensormessdaten zu quasikontinuierlichen Spuren einzelner Objekte verstanden. Die Schritte der Detektionsextraktion und Bündelung eines Radarsensors, welche in Kapitel 2.1.4 mit der Signalverarbeitungskette vorgestellt wurden, ergeben Objekthypothesen, welche nur für den aktuellen Zyklus gültig sind. Im Tracking-Algorithmus erfolgt nun die Assoziation zu Objekthypothesen vorheriger Messzyklen.⁴⁵ Dazu werden die vorherigen Zustände der Objekthypothesen für den aktuellen Messzyklus prädiziert, sodass durch unterschiedliche Assoziationsmethoden eine Zuordnung der Messdaten erfolgen kann. Die Objektzustände beschreiben in der Regel Position und Geschwindigkeit des Objektes, können zusätzlich aber auch die Orientierung oder Objektdimensionen abdecken. Herausforderungen im Tracking sind durch die unbekannte Zahl an Objekten und deren genauso unbekannten Bewegungen gegeben. Zusätzlich unterliegt das Tracking-Ergebnis dabei folgenden drei Unsicherheiten:⁴⁶

1. Zustandsunsicherheit: Die Zustandsunsicherheit beschreibt die Unsicherheit in den physikalischen Messgrößen des Sensors und ist eine direkte Folge der bei Sensoren nicht vermeidbaren Messfehlern, welche auch unter Messrauschen bekannt sind.
2. Existenzunsicherheit: Die Existenzunsicherheit beschreibt die Unsicherheit, ob ein von der Sensorik erfasstes Objekt⁴⁷ tatsächlich real vorhanden ist. Diese Unsicherheit ergibt sich unter anderem aufgrund von Störsignalen, welche Detektionen ohne ein vorhandenes Objekt darstellen.

⁴⁴ Granström, K. et al.: Introduction to Extended Object Tracking (2016), S. 1.

⁴⁵ Winner, H.: Radarsensorik (2015), S. 293.

⁴⁶ Dietmayer, K.: Prädiktion von maschineller Wahrnehmungsleistung beim automatisierten Fahren (2015), 425f.

⁴⁷ „Objekt“ ist hier als erfasstes Element im Sensorfeld zu verstehen, dessen Relevanz für das Egofahrzeug noch nicht identifiziert ist.

-
3. Klassenunsicherheit: Mit Klassenunsicherheit ist die Unsicherheit bezüglich der korrekten Zuordnung gemeint, welche sich aus den verwendeten Verfahren zur Datenassoziation im Tracking-Algorithmus ergibt.

Für Applikationen, welche auf einer Umfoldsensorik basieren, ist es daher notwendig, diese Unsicherheiten und Objektbewegungen zuverlässig zu erkennen und, wenn möglich, präzisieren zu können. Dafür eignet sich die nachfolgende Herleitung des Satz von Bayes unter Anwendung auf Tracking-Algorithmen, welcher der Arbeit von Challa, S. et al. entnommen ist.⁴⁸ Der Ursprung dessen ist die von Thomas Bayes im Jahr 1763 vorgestellte Bayes-Gleichung zur Berechnung von bedingten Wahrscheinlichkeiten:

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)} \quad (4-1)$$

Mit $p(x|y)$ wird die Wahrscheinlichkeit ausgedrückt, dass Ereignis x eintritt, wenn zuvor Ereignis y eingetreten ist. Für $p(y|x)$ gilt die umgekehrte Ereignisreihenfolge. Die Größen $p(y)$ und $p(x)$ beschreiben jeweils die totalen Wahrscheinlichkeiten für das Eintreten der einzelnen Ereignisse.

Wendet man die Bayes-Gleichung nun auf das Tracking-Problem an, sind x der zu schätzende Objektzustand und y die zu diesem Objekt gehörende Messwerte, die der Sensor liefert. Da der Zustand x nicht bekannt ist und Unsicherheiten unterliegt, wird das Wissen über diese Größe als eine Wahrscheinlichkeitsdichtefunktion $p(x)$ ausgedrückt. Dieses Wissen $p(x)$ kann mit jedem Messzyklus konsekutiv aktualisiert werden, da durch weitere Sensordaten y sich der Umfang an Informationen über umliegende Objekte erhöht. Die Aktualisierung dieser Wahrscheinlichkeitsdichtefunktion des Objektzustands erfolgt unter Anwendung des Satz von Bayes, welcher die bedingte Wahrscheinlichkeitsdichtefunktion $p(x|y)$ liefert – die sogenannte A-posteriori-Wahrscheinlichkeit für den Objektzustand x , nachdem die Größen y gemessen wurden. Diese ergibt sich aus der Multiplikation der A-priori-Wahrscheinlichkeit $p(x)$ mit der bedingten Wahrscheinlichkeit $p(y|x)$. Die Größe $p(y|x)$ ist die sogenannte Likelihood-Funktion, welche die Wahrscheinlichkeit beschreibt, dass die Messwerte y detektiert werden, wenn der Objektzustand x gilt. Die totale Wahrscheinlichkeit $p(y)$ dient zum Normieren des Wertes.

Die Gleichung (4-1) lässt sich für einen Zeitpunkt k nun wie folgt ausdrücken:

$$p(\mathbf{S}^k|\mathbf{y}^k) = \frac{p(\mathbf{y}^k|\mathbf{S}^k)p(\mathbf{S}^k)}{p(\mathbf{y}^k)} \quad (4-2)$$

Dabei ist \mathbf{S}^k ein Vektor, welcher aller bisher geschätzten Objektzustände bis zum Zeitpunkt k enthält. \mathbf{y}^k enthält analog alle Messdaten und lässt sich schreiben als $\mathbf{y}^k = (\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_k) = (\mathbf{y}^{k-1}, \mathbf{y}_k)$, wobei \mathbf{y}_i die Messung zum Zeitpunkt i beschreibt. Mit der Gleichung (4-2) würde die A-posteriori-Wahrscheinlichkeit für \mathbf{S}^k unter Berücksichtigung aller bisherigen Zustände bis zum Zeitpunkt k berechnet werden. Dies entspricht allerdings nicht der Realität, da folgenden Annahmen gelten:

⁴⁸ Challa, S. et al.: Fundamentals of Object Tracking (2011), S. 5–9.

- Messdaten \mathbf{y}^{k-1} zum Zeitpunkt $k - 1$ sind unabhängig von Objektzuständen \mathbf{S}^k zum Zeitpunkt $\geq k$,
- Messdaten \mathbf{y}_k sind unabhängig von Messdaten \mathbf{y}^{k-1} vorheriger Zeitpunkte $\leq k - 1$,
- Messdaten \mathbf{y}_k sind unabhängig von Objektzuständen \mathbf{S}^{k-1} vorheriger Zeitpunkte $\leq k - 1$,

Mit diesen Annahmen lässt sich (4-2) zur rekursiven Bayes-Schätzung umformen:

$$p(\mathbf{S}^k | \mathbf{y}^k) = \frac{p(\mathbf{y}_k | \mathbf{S}_k)}{p(\mathbf{y}_k | \mathbf{y}^{k-1})} p(\mathbf{S}_k | \mathbf{S}_{k-1}) p(\mathbf{S}^{k-1} | \mathbf{y}^{k-1}) \quad (4-3)$$

Um nun die A-posteriori-Wahrscheinlichkeit $p(\mathbf{S}_k | \mathbf{y}^k)$ für einen Objektzustand \mathbf{S}_k zum Zeitpunkt k zu ermitteln, wird die Gleichung (4-3) nach jedem Objektzustand \mathbf{S}_i integriert. Daraus folgt die bedingte Zustandswahrscheinlichkeit in (4-4).

$$p(\mathbf{S}_k | \mathbf{y}^k) = \frac{1}{p(\mathbf{y}_k | \mathbf{y}^{k-1})} p(\mathbf{y}_k | \mathbf{S}_k) \int_{\mathbf{S}_{k-1}} p(\mathbf{S}_k | \mathbf{S}_{k-1}) p(\mathbf{S}_{k-1} | \mathbf{y}^{k-1}) d\mathbf{S}_{k-1} \quad (4-4)$$

Das Integral ist als Chapman-Kolmogorov-Gleichung bekannt und liefert den prädierten Objektzustand \mathbf{S}_k unter Berücksichtigung aller Messdaten \mathbf{y}^{k-1} bis zum Zeitpunkt $k - 1$ und dem Objektzustand \mathbf{S}_{k-1} zum Zeitpunkt $k - 1$. Sobald mit einem neuen Zyklus die Messdaten \mathbf{y}^k vorhanden sind, kann der zuvor prädierte Objektzustand mit der Likelihood-Funktion $p(\mathbf{y}_k | \mathbf{S}_k)$ korrigiert und durch $p(\mathbf{y}_k | \mathbf{y}^{k-1})$ erneut normiert werden. Das Lösen dieser rekursiven Funktion ist der Kern aller Tracking-Algorithmen.⁴⁹

Zur Anwendung des Satz-von-Bayes auf Objektzustände und Messdaten werden zuvor das Bewegungs- und Sensormodell des Tracking-Problems aufgestellt. Dadurch wird problemspezifisches Modellwissen über Bewegungseinschränkungen, Vorzugsrichtungen und weitere vorhandene Objektinformationen in die Zustandsschätzung eingebracht.⁵⁰ Der Zustand \mathbf{S} eines erfassten Objekts wird in der Regel durch Größen wie Position, Geschwindigkeit, Beschleunigung oder weitere kinematische Eigenschaften beschrieben.⁴⁹ Das Bewegungsmodell ergibt sich damit aus einer zeitvarianten Funktion, welche die Zustandsänderung zwischen zwei konsekutiven Messzyklen schätzt und ebenfalls unter dem Namen Zustandsübergangsfunktion bekannt ist. Da diese Schätzung nicht exakt ist und Modellfehler existieren, wird eine Störgröße, das Modellrauschen, hinzugefügt. Das Bewegungsmodell ergibt sich nachfolgend zu:

$$\mathbf{S}_k = f(\mathbf{S}_{k-1}, \mathbf{v}_k) \quad (4-5)$$

$f(\cdot)$ stellt dabei die Zustandsübergangsfunktion dar, welche linear oder nicht-linear sein kann, während \mathbf{v}_k das Modellrauschen beschreibt.

Das Sensormodell hingegen gibt an, wie Messdaten und Zustandsgrößen zusammenhängen und ermöglicht so die Berechnung der Likelihood-Funktion. Eine prädierte Zustandsschätzung \mathbf{S}_k kann so mit nun vorhandenen Messdaten aktualisiert werden. Damit ergibt sich das Sensormodell aus einer

⁴⁹ Challa, S. et al.: Fundamentals of Object Tracking (2011), S. 16.

⁵⁰ Munz, M.: Zustands- und Existenzschätzung für die Fahrzeugumfeldererkennung (2011), S. 13.

Messfunktion $h(\mathbf{S}_k)$. Kann eine Zustandsgröße beispielsweise direkt gemessen werden, ist $h(\cdot)$ eine Eins-zu-eins-Abbildung.

Da Fehler in Messdaten nicht auszuschließen sind, enthält das Sensormodell ebenfalls eine Störgröße und ergibt sich zu

$$\mathbf{y}_k = h(\mathbf{S}_k, \mathbf{w}_k), \quad (4-6)$$

wobei \mathbf{w}_k das Sensorrauschen abbildet.^{51a}

Die Berechnung der A-posteriori-Wahrscheinlichkeit in (4-4) lässt sich damit unter Berücksichtigung von Bewegungs- und Sensormodell in zwei Schritte unterteilen – dem Prädiktions- und dem Innovationsschritt. Der Prädiktionschritt der Objektzustände unter Berücksichtigung des Bewegungsmodell in (4-5) geschieht in der in (4-4) enthaltenen Chapman-Kolmogorov-Gleichung. Die darauffolgende Innovation, welche die Prädiktion aktualisiert bzw. korrigiert, geschieht durch Einbringen der Likelihood-Funktion $p(\mathbf{y}_k|\mathbf{S}_k)$, welche das Sensormodell aus (4-6) nutzt. Die Likelihood-Funktion ist in (4-4) der Chapman-Kolmogorov-Gleichung vorangestellt.

Dieses Schätzverfahren durch Prädiktion und Innovation ist als allgemeiner Bayes-Filter bekannt und bildet die Grundlage für alle heute üblichen Methoden der stochastischen Zustandsschätzung. Dieser Filter ist in seiner allgemeinen Form allerdings nicht effizient zu implementieren.^{51b} Daher werden in Kapitel 4.7. unterschiedliche Anwendungen von Filtern unter Einordnung in der Systemarchitektur eines Tracking-Algorithmus erläutert.

4.2 Tracking-Arten

Alle Tracking-Probleme lassen sich in insgesamt fünf Untergruppen aufteilen, welche die Wahl der Methoden für die einzelnen Schritte im Tracking-Algorithmus einschränkt. Diese Gruppen sind nicht alle gegensätzlich, da unterschiedliche Aspekte betrachtet werden wie einerseits die Zahl der zu trackenden Objekte und andererseits deren geometrische Ausdehnung.

Hinsichtlich der Objektzahl gibt es das sogenannte Single-Object Tracking und Multi-Object Tracking. Single-Object Tracking (SOT) nutzt die Annahme, dass zu jeder Zeit genau ein Objekt im Sichtfeld des Sensors vorhanden ist, während Multi-Object-Tracking (MOT) kein Wissen über die Zahl der Objekte hat. Damit beschreibt MOT ein signifikant komplexeres Problem, da die Zahl der hypothetischen Datenassoziationen, also die Zuordnung von Messdaten zu prädictierten Objektzuständen, nicht auf ein zu trackendes Objekt beschränkt ist.⁵²

Bezüglich der geometrischen Ausdehnung der Objekte, lassen sich Tracking-Probleme in Point-Object Tracking (POT), Extended-Object Tracking (EOT) und Group-Object Tracking (GOT) einteilen. POT unterliegt der Annahme, dass jedes Objekt maximal eine Detektion pro Messzyklus erzeugt und damit nur eine Auflösungszelle durch das Objekt besetzt ist, wohingegen EOT annimmt, dass ein Objekt mehrere Detektionen pro Zyklus erzeugt und so mehrere Auflösungszellen belegt. So können

⁵¹ Dietmayer, K.: Prädiktion von maschineller Wahrnehmungsleistung beim automatisierten Fahren (2015), a: S. 427–428; b: S. 428.

⁵² Granström, K.; Svensson, L.: Introduction to Single-Object Tracking (2019).

die Objektdimensionen abgeschätzt werden, während im POT keine Informationen dazu vorliegen. GOT erweitert das EOT-Konzept und fasst Objekte ähnlicher Zustände zu einer einzigen Entität zusammen. Damit werden die zur Entität gehörenden Subobjekte nicht einzeln verfolgt, sondern als Gruppe. Die Wahl zwischen diesen drei Tracking-Arten hängt damit primär von den vorliegenden Sensoreigenschaften, insbesondere dem Auflösungsvermögen ab.⁵³

Da mit dem zu implementierenden Tracking-Algorithmus Fahrsituationen mit einer unbekannten Anzahl an Verkehrsteilnehmern verarbeitet werden, liegt der Fokus im weiteren Verlauf der Arbeit auf Multi-Object-Tracking. Hinsichtlich der Objektdimensionen gilt das Auflösungsvermögen der verwendeten Radarsensoren als Einschränkungen. Werden mehrere Detektionen von einem Objekt erzeugt, können dieser zur Schätzung der Objektdimensionen genutzt werden, allerdings ist dies nicht kontinuierlich der Fall, sodass der zu implementierende Tracking-Algorithmus weder der Annahme des POT noch der Annahme des EOT entspricht. Die Methode zur Schätzung der Objektdimensionen ist in Kapitel 6.6.5 dargestellt.

4.3 Systemarchitektur

Um aus Sensordaten eine Objektliste auszugeben, folgt ein Tracking-Algorithmus den in Abbildung 4-1 dargestellten Schritten. Dieses Kapitel erläutert die abgebildeten Schritte und stellt anschließend unterschiedliche Methoden zur Umsetzung jedes einzelnen Schrittes vor. Da der zu entwickelnde Algorithmus auf Sensordaten in Form von Detektionen aus der Signalverarbeitungskette basiert, werden diese hier als Eingangsdaten gezeigt.

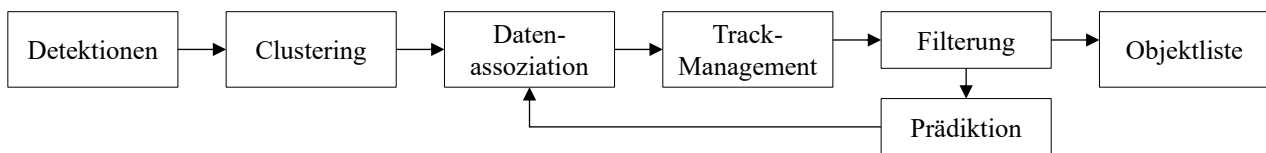


Abbildung 4-1: Systemarchitektur eines Tracking-Algorithmus⁵⁴

Bevor der Tracking-Algorithmus die Detektionen verarbeitet, besteht die Möglichkeit einen optionalen Schritt zur Filterung der Radarsignle durchzuführen. So können Reflexionen, die von keinem Objekt stammen, wie beispielsweise Straßenreflexionen, anhand der Signalleistung entfernt werden.

Ist die Datenvorverarbeitung abgeschlossen erfolgt das Clustering. Hier werden Mehrfachreflexionen, die von einem Objekt stammen zum sogenannten Cluster zusammengefasst. Das Clustering sucht dabei nach Ähnlichkeiten zwischen einzelnen Detektionen einer Messung und fasst diese zusammen, sodass der Algorithmus im weiteren Verlauf nicht jede Einzeldetektion verarbeiten muss.⁵⁴

Im nächsten Schritt der Datenassoziaton werden die zuvor gebildeten Cluster bereits bestehenden Tracks von erfassten Objekten vorheriger Zyklen zugeordnet. Dazu werden die Distanzen zwischen

⁵³ Granström, K. et al.: Introduction to Extended Object Tracking (2016), S. 3.

⁵⁴ Manjunath, A. et al.: Radar Based Object Tracking (2018), S. 2.

Cluster und Objekt berechnet. Liegen diese unter einem zuvor definierten Schwellenwert, ordnet die Datenassoziation sie dem entsprechenden Objekt zu.⁵⁵

Das Track-Management verwaltet die einzelnen Objekte. Ein Track kann unterschiedliche Status haben, welche sich aus dem Ergebnis der Datenassoziation des aktuellen Zyklus und vorherigen Zyklen ergibt. Wurde einem Objekt beispielsweise keine Detektion zugeordnet, wird dieser Track auf die Löschung vorbereitet und schließlich gelöscht, falls in einer vorher festgelegten Zahl von konsekutiven Messungen ebenfalls keine Datenassoziation erfolgte. Genauso erfolgt hier die Initiierung von neuen Tracks aus Detektionen, die keinem Objekt zugeordnet werden konnten.⁵⁶

Erfolgte für einen prädierten Objektzustand eine Datenassoziation, wird mit diesen neu vorliegenden Messdaten die Prädiktion korrigiert. Kapitel 4.1 stellte diesen Schritt als sogenannte Innovation bereits vor. Unterschiedliche Filtermethoden, wie das Kalman Filter oder das Partikel-Filter, ermöglichen diesen Schritt unter Verwendung des zuvor definierten Bewegungs- und Sensormodell. Der korrigierte Objektzustand wird daraufhin zur Prädiktion und in der Datenassoziation des kommenden Zyklus verwendet und ebenfalls in der Objektliste ausgegeben.⁵⁷

4.4 Clustering

Wenn bekannt ist, dass von einem Objekt mehr als eine Detektion empfangen werden kann, bietet das Clustern die Möglichkeit diese Mehrfachdetektionen zusammen zu fassen und so den Rechenaufwand im Schritt der Datenassoziation zu minimieren. Dies ist insbesondere im Extended-Object Tracking vorteilhaft. Unterschiedliche Ansätze werden unter anderem von Hammarstrand et al.⁵⁸, Broßheit et al.⁵⁹ und Kellner et al.⁶⁰ vorgestellt.

Hammarstrand et al. berücksichtigen dabei die Karosserieform und seine Reflexionseigenschaften. Ein Radarsensor erhält demnach eine Vielzahl seiner Reflexionen von sogenannten Reflexionszentren, welche unter anderem an den Radkästen oder im Scheinwerferbereich liegen. In Abbildung 4-2 sind die Reflexionszentren und ihre konischen Sichtbarkeitsbereiche dargestellt. Eine Reflexion kann demnach nur empfangen werden, wenn der Radarsensor sich im jeweiligen Sichtbarkeitsbereich befindet und ein Punkt des Reflexionszentrums auf der Normalenfläche zu den Radarstrahlen liegt.

⁵⁵ Danielsson, L.: Diss., Tracking and radar sensor modelling (2010), S. 59.

⁵⁶ Hyun, E.; Lee, J.-H.: Track Management Table, S. 2.

⁵⁷ Challa, S. et al.: Fundamentals of Object Tracking (2011), S. 22.

⁵⁸ Hammarstrand, L. et al.: Extended Object Tracking using a Radar Resolution Model (2012).

⁵⁹ Broßheit, P. et al.: Shape Estimation for Extended Object Tracking.

⁶⁰ Kellner, D. et al.: Grid-based DBSCAN for clustering (2012).

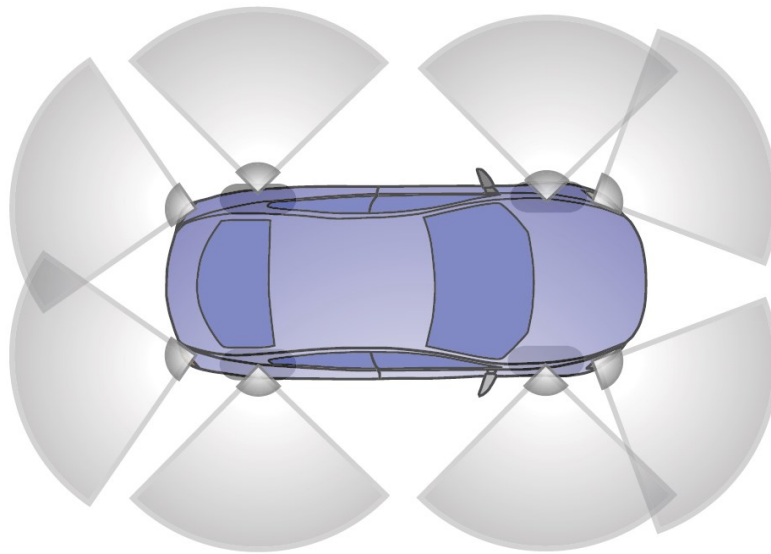


Abbildung 4-2: Reflexionszentren an einer Fahrzeugkarosserie⁵⁸

Ein weiterer Ansatz wird von Broß et al. geliefert. Die Autoren erweitern den Ansatz einer „Orientated Bounding-Box“, indem die Unsicherheiten der Messdaten integriert werden. Die Form des Objekts wird aus den Mehrfachreflexionen extrahiert und konsekutiv aktualisiert. Die Aktualisierung erfolgt unter Berücksichtigung der Korrelation zwischen Positions- und Größenänderung der Bounding-Box. Untersuchungen dieser probabilistischen Methode durch die Autoren zeigten, dass ein mittleres Abweichungsquadrat (engl.: Root Mean Square Error) für die Position kleiner als 0,1 m und für die Orientierung kleiner als $1,7^\circ$ erreichbar ist.

Kellner et al. zeigen in ihrer Arbeit eine Abwandlung des Density-based spatial Clustering of Applications with Noise Algorithmus (DBSCAN) für Extended-Object Tracking, welcher erstmals von Ester et al.⁶¹ vorgestellt wurde. Der DBSCAN-Algorithmus untersucht Reflexion hinsichtlich zwei Parameter, Entfernung und Azimutwinkel, um aus diesen Mehrfachreflexionen zusammenzufassen. Diese Methode zeichnet sich durch seine Berücksichtigung von Störsignalen aus, d.h. Reflexionen werden entweder einem Cluster zugeordnet oder als Rauschen klassifiziert und so nicht als relevantes Objekt eingestuft. Da die Entfernungs- und Azimutwinkelauflösung des Sensors konstant sind, nimmt die erreichbare Dichte der Reflexionen mit der Entfernung stark ab. Ein Fahrzeug in $r = 3$ m erzeugt damit eine deutlich höhere Zahl an Reflexionen als das gleiche Fahrzeug in $r = 80$ m wie in Abbildung 4-3 beispielhaft dargestellt ist. Kellner et al. erweiterten diese Methode zum Grid-based DBSCAN auf Basis eines hochauflösenden Radarsensors. Hierbei werden statt der Entfernung und des Azimutwinkels die bereits umgerechneten x- und y-Positionen im kartesischen Koordinatensystem genutzt. Weitere im Algorithmus genutzte Parameter, wie der Suchradius, werden für jeden Punkt im kartesischen Koordinatensystem individuell berechnet. Diese integrierte Abhängigkeit führt zu einem genaueren Clusterergebnis bezüglich der Vernachlässigung von Störsignalen, der Trennfähigkeit von Objekten und der geschätzten Objektorientierung.

⁶¹ Ester, M. et al.: DBSCAN for discovering clusters (1996).

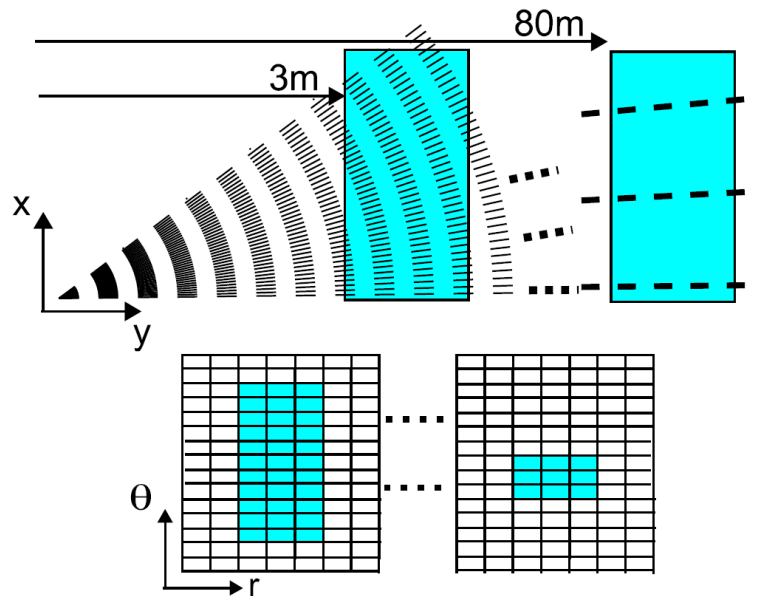


Abbildung 4-3: Unterschiedliche Reflexionsdichte eines Fahrzeugs in Abhängigkeit der Entfernung in kartesischen Koordinaten (oben) und im Azimutwinkel-Entfernungs-Gitter (unten)⁶²

Alle bisherigen genannten Clusterverfahren liefern zusätzliche Informationen aufgrund der ermittelten Objektform und -orientierung, erfordern dafür aber eine Vielzahl an Reflexionen pro Objekt und sind damit nur für Tracking-Algorithmen geeignet, welche Eingangsdaten aus einem hochauflösenden⁶³ Radarsensor erhalten. Da dies mit dem beim FZD vorhandenen Radarsensor nicht der Fall ist, sind diese Clusterverfahren innerhalb dieser Arbeit nicht umsetzbar.

Eine Methode, welche effektiv bei einer kleineren Anzahl an Reflexionen pro Objekt ist, bietet der sogenannte k-Means-Algorithmus. Dieser fasst Reflexionen ähnlicher Position und Geschwindigkeit zu einem Cluster zusammen. Die Ähnlichkeit wird über einen zuvor festgelegten Schwellenwert entschieden. Aus den zugeordneten Reflexionen wird über deren Mittelwert der Objektzustand ermittelt.⁶⁴ Aufgrund der geringeren Zahl an Reflexionen ist diese Methode mit einem geringen Rechenaufwand verbunden, ermittelt allerdings keine Informationen zur Objektform und -orientierung. Die Zahl der vorhandenen Objekte muss beim k-Means-Algorithmus vor Anwendung bekannt sein, so dass diese Methode für den Einsatz von komplexen Fahrsituation mit variierender Objektzahl nicht geeignet ist und so als Clusterverfahren innerhalb dieser Arbeit nicht umsetzbar ist.

Das für den zu implementierenden Tracking-Algorithmus genutzte Clusterverfahren basiert auf einer Implementierung von Ackermann⁶⁵, welcher ein rekursives Verfahren zur Generierung einer unbekannten Zahl von Clustern nutzt. Mit diesem Verfahren werden Detektionen nicht nur aufgrund ihrer Position zusammengefasst, sondern zusätzlich anhand ihrer Relativgeschwindigkeit unterschieden. So wird sichergestellt, dass das Clustering beispielsweise Detektionen von zwei entgegengesetzt fahrenden Objekten nicht zusammengefasst, wenn sie die gleiche Entfernung zum Egofahrzeug haben.

⁶² Kellner, D. et al.: Grid-based DBSCAN for clustering (2012), S. 366.

⁶³ Als hochauflösend sind hier Sensoren gemeint, welche stets mehr als eine Detektion pro Objekt liefern.

⁶⁴ Wood, T. M.: Diss., Random Finite Sets (2011), S. 86–87.

⁶⁵ Stefan Ackermann: Masterthesis, Systematische Untersuchung von Radar Tracking-Algorithmen (2017), S. 40–41.

Um Ähnlichkeiten in der Position und Relativgeschwindigkeit der Reflexionen zu ermitteln, werden vor Schwellenwerte definiert. Eine detaillierte Erläuterung dieses Verfahrens ist in Kapitel 6.3 gegeben.

4.5 Datenassoziation

Die Datenassoziation ordnet die Messdaten prädizierten Objektzuständen zu. Die Herausforderung im Multi-Object Tracking liegt in der Zuordnung von Messdaten zu Objekten, deren prädizierten Zustände so nah beieinander liegen, dass sich die Suchbereiche überschneiden. Ein solches Beispiel ist in Abbildung 4-4 dargestellt. Darauf sind die prädizierten Objektzustände $\hat{\mathbf{y}}_{k|k-1}^1$ und $\hat{\mathbf{y}}_{k|k-1}^2$ und die zum aktuellen Messzyklus gehörenden Messwerte \mathbf{y}_k^i mit $i = [1, 2, \dots, 7]$ zu sehen. Der Messwert \mathbf{y}_k^6 liegt dabei in beiden Suchbereichen. Eine eindeutige Zuordnung ist damit nicht direkt möglich.

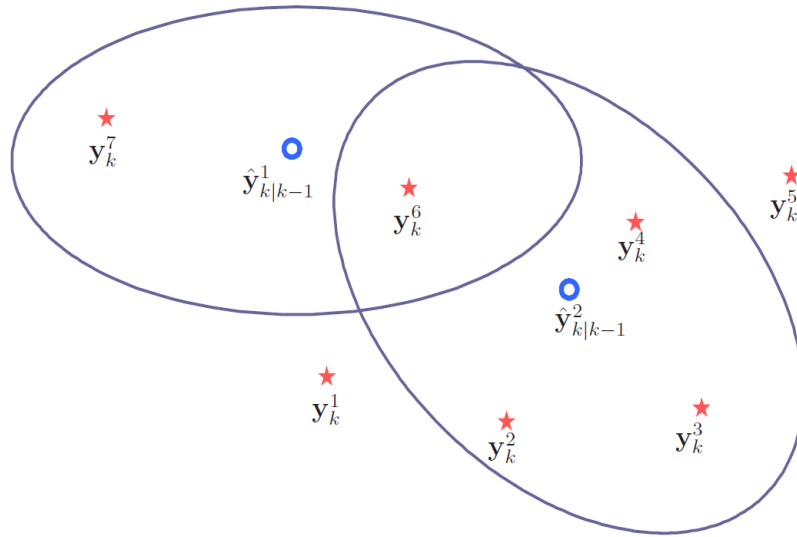


Abbildung 4-4: Mögliches Szenario zur Datenassoziation mit prädizierten Objektzuständen und Messdaten⁶⁶

4.5.1 Probabilistic Data Association und Erweiterungen

Eine Methode zur Datenassoziation, welche alle Messwerte, die in den Suchbereich eines Objekts fallen, berücksichtigt, wurde 1975 von Bar-Shalom und Tse unter dem Namen Probabilistic Data Association (PDA)⁶⁷ vorgestellt und wenige Jahre später zur Joint Probabilistic Data Association (JPDA)⁶⁸ weiterentwickelt.

Die Grundidee beider Methoden ist die Gewichtung aller Messdaten im Suchbereich. Dazu werden Zugehörigkeitswahrscheinlichkeiten der Messwerte zu den Objekten unter Berücksichtigung der Unsicherheiten im Bewegungs- und Sensormodell ermittelt. Der Unterschied zwischen beiden Methoden liegt in der Zahl der Objekte, welche verfolgt werden kann. Das PDA geht dabei von nur einem

⁶⁶ Danielsson, L.: Diss., Tracking and radar sensor modelling (2010), S. 62.

⁶⁷ Bar-Shalom, Y.; Tse, E.: Tracking in a Cluttered Environment With Probabilistic Data Association (1975).

⁶⁸ Fortmann, T. et al.: Sonar tracking of multiple targets using joint probabilistic data association (1983).

Objekt im Sichtfeld des Sensors aus, sodass die Anwendung im Multi-Object Tracking nicht empfohlen ist, da insbesondere Objekte, deren Trajektorien sich kreuzen, nicht effektiv verarbeitet werden. Bei einer Implementierung wäre damit nicht ausgeschlossen, dass ein Messwert mehreren Objekten zugeordnet wird.

Das JPDA hingegen betrachtet mehrere zu verfolgende Objekte und den gegenseitigen Einfluss der Messwerte, sodass für eine Zuordnung die größte gemeinsame Zugehörigkeitswahrscheinlichkeit genutzt wird. Bei der Anwendung dieser Methode wird allerdings vorausgesetzt, dass die Zahl der Objekte stets bekannt ist.⁶⁹ Somit ist dieses Verfahren zur Anwendung bei komplexeren Fahrsituationen mit einer variierenden Zahl an Objekten nicht umzusetzen. Eine Erweiterung dieses Verfahrens bietet das Joint Integrated Data Association (JIPDA), welches von Mušicki und Evans vorgestellt wurde.⁷⁰ Diese Methode integriert die Existenzwahrscheinlichkeiten der Objekte und aktualisiert diese konsekutiv, während das JPDA diese ungeachtet lässt und von keiner Unsicherheit der Existenz ausgeht. Die Berechnung aller Zugehörigkeitswahrscheinlichkeiten erfordert das Aufstellen eines Hypothesenbaums, welcher mit steigender Objekt- und Clusterzahl wächst und so einen erhöhten Rechenaufwand mit sich zieht.⁷¹

4.5.2 Multiple-Hypothesis Tracking

Einen ähnlichen probabilistischen Ansatz liefert Reid mit dem sogenannten Multiple-Hypothesis Tracking (MHT). Hier ergeben sich die Zugehörigkeitswahrscheinlichkeiten der Messwerte nicht ausschließlich aus dem aktuellen Zyklus, sondern werden mit jedem Zyklus konsekutiv aktualisiert. Die Entscheidung der Datenassoziation findet damit verzögert statt.⁷² Das Verfahren lässt sich mit einem Hypothesenbaum anschaulich darstellen, der mit jedem Zyklus um eine Knotenebene erweitert wird. Das MHT betrachtet für alle Messwerte je Zyklus drei Wahrscheinlichkeiten, die je einen Ast des Hypothesenbaums darstellen.⁷³

1. Die Messdaten gehören zu einem verfolgten Objekt.
2. Die Messdaten gehören zu einem neuen Objekt.
3. Die Messdaten gehören zu keinem Objekt.

Fällt die Wahrscheinlichkeit einer Hypothese unter einen zuvor definierten Schwellenwert, wird die Hypothese aussortiert und nicht mehr weitergeführt.⁷⁴ Im Sinne der Vollständigkeit der Assoziationen stellt das MHT-Verfahren eine optimale Methode dar und ist damit stabiler bezüglich Fehlklassifikationen. Allerdings resultiert das konsekutive Aufstellen und Aktualisieren der Hypothesen in einem sehr rechenaufwendigen Verfahren, da die Anzahl der Hypothesen exponentiell steigt. Zusätzlich ist,

⁶⁹ Bar-Shalom, Y. et al.: The Probabilistic Data Association Filter (2009), 91.

⁷⁰ Musicki, D.; Evans, R.: Joint integrated probabilistic data association: JIPDA (2004).

⁷¹ Schuster, M.: Diss., Multiple Object Tracking using JIPDA (2017), S. 27.

⁷² Reid, D.: An algorithm for tracking multiple targets (1979).

⁷³ Liu, F.: Diss., Objektverfolgung durch Sensordatenfusion (2009), S. 46.

⁷⁴ Kim, C. et al.: Multiple Hypothesis Tracking Revisited (2015 - 2015), S. 4700.

im Vergleich zu den restlichen genannten Verfahren, die Implementierung signifikant komplexer und kann „bis zu sechs Personenmonate“ dauern.⁷⁵

4.5.3 Nearest Neighbor und Global Nearest Neighbor

Die folgende Methode betrachtet im Gegensatz zu bisher genannten Verfahren nicht alle möglichen Hypothesen der Zugehörigkeitswahrscheinlichkeiten, sondern nur diejenige mit der größten Wahrscheinlichkeit, und ist daher unter dem Namen Nearest Neighbor (NN) bekannt. Dieses Verfahren ordnet den Objekten die Messwerte mit der geringsten Distanz zu. Die Wahl der Berechnungsmethode, ob euklidische Distanz oder Mahalanobis-Distanz, ist nicht festgelegt. Da diese Methode jedes Objekt separat betrachtet und so nur die lokale optimale Assoziation untersucht, besteht die Möglichkeit, dass ein Messwert mehreren Objekten zugeordnet wird. Um ein globales Optimum der Assoziation zu erreichen, empfiehlt sich die Verwendung des Verfahrens Global Nearest Neighbor (GNN). Hier werden alle vorhandenen Distanzen simultan betrachtet und Mehrfachzuordnungen vermieden.⁷⁶ Die unterschiedlichen Ergebnisse beider Verfahren sind in Abbildung 4-5 dargestellt. Das NN-Verfahren ist links, das GNN-Verfahren rechts dargestellt. Man erkennt, dass im GNN-Verfahren keine Mehrfachzuordnung des Messwertes y_k^2 erfolgte. Durch das Betrachten einer einzelnen Hypothese je Objekt, sind das NN- und GNN-Verfahren sehr recheneffizient und effektiv in simplen Szenarien mit einem hohen Signal-Rausch-Verhältnis. Bei komplexeren Szenarien besteht durch die Vernachlässigung der Unsicherheiten die Möglichkeit von Fehlklassifikationen.⁷⁷

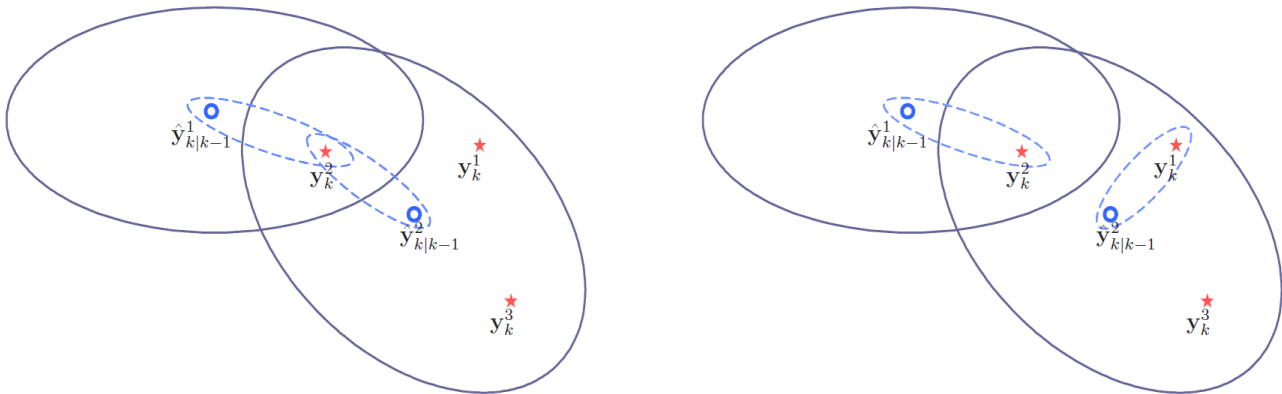


Abbildung 4-5: Unterschiedliche Ergebnisse der Datenassoziation nach den Verfahren des Nearest Neighbor (links) und Global Nearest Neighbor (rechts)⁷⁸

⁷⁵ Blackman, S.; Popoli, R.: Design of tracking systems (1999), S. 373., nach Wood, T. M.: Diss., Random Finite Sets (2011), S. 37.

⁷⁶ Wood, T. M.: Diss., Random Finite Sets (2011), S. 38.

⁷⁷ Liu, F.: Diss., Objektverfolgung durch Sensordatenfusion (2009), S. 46.

⁷⁸ Danielsson, L.: Diss., Tracking and radar sensor modelling (2010), S. 64.

4.5.4 Random Finite Sets

Ein anderer Ansatz basiert auf Random Finite Sets (RFS), welcher erstmals von Mahler⁷⁹ vorgestellt wurde. RFS fassen die gesamte beobachtete Situation bzw. Objekte und Messwerte als jeweils individuelle Zustände auf. Das Problem der Datenassoziation wird damit umgangen, da die Zustände mathematisch mithilfe der Finite Set Statistics zusammengeführt werden. Die Zahl der Objekte und deren Zustände werden so mit Berücksichtigung bedingter Wahrscheinlichkeiten aus den Messdaten ermittelt.⁸⁰ Anwendung fanden RFSs in Implementierungen von Probability Hypothesis Density Filter (PHD) durch Mahler⁸¹ oder Multi-Target Multi-Bernoulli Filter durch Vo et al.⁸² Aufgrund der Komplexität des Verfahrens eignen sich RFS aktuell nicht für Echtzeitanwendung. Dieses Problem wird in neueren Implementierungen von Hu et al.⁸³ und Reuter et al.⁸⁴ optimiert.

Zusammenfassend ist festzuhalten, dass die Verfahren PDA, JPDA und JPDA sich durch ihre Berücksichtigung von Fehlklassifikation auszeichnen, wobei allerdings die Objektzahl zu Beginn bekannt sein muss. Dieses Problem ist beim MHT nicht vorhanden. Dieses Verfahren ist jedoch aufgrund der relativ niedrigen Rentabilität bezüglich des Implementierungsaufwands als Datenassoziationsmethode für den zu implementierenden Algorithmus nicht geeignet. Ein ähnliches Ergebnis liefern die Random Finite Sets. Durch das Umgehen der Datenassoziation liefert diese Methode stabile Tracking-Ergebnisse. Die Anwendung der Finite Set Statistics und der damit verbundenen Komplexität sind im Rahmen dieser Arbeit allerdings nicht umzusetzen. Das NN- und GNN-Verfahren zeichnen sich durch eine niedrige Komplexität und hoher Rentabilität bezüglich des Implementierungsaufwands sowie hoher Recheneffizienz aus. Daher eignen sich beide Verfahren zur Implementierung innerhalb dieser Arbeit. Um Mehrfachzuordnung zu vermeiden, wird das GNN-Verfahren zu Datenassoziation für den zu entwickelnden Tracking-Algorithmus ausgewählt.

4.6 Track-Management

Das Track-Management verwaltet die Status der verfolgten Objekte. Neue Objekte befinden sich in der Initiierungsphase, bevor sie nach einigen Zyklen und erfolgreich zugeordneten Messdaten als betätigtes Objekt klassifiziert werden. Analog verläuft die Löschung von Objekten. Wird dem Objekt kein Messwert zugeordnet, erfolgt keine Aktualisierung des Objektzustands durch Sensordaten. Der Objektzustand basiert dann ausschließlich auf der Prädiktion. Erfolg für einige Zyklen keine Datenassoziation, wird das Objekt schließlich gelöscht.

⁷⁹ Mahler, R.P.S.: "Statistics 101" for multisensor, multitarget data fusion (2004).

⁸⁰ Clark, D. E.: Diss., Tracking with the Probability Hypothesis Density Filter (2006), S. 11.

⁸¹ Mahler, R.P.S.: Multitarget bayes filtering via first-order multitarget moments (2003).

⁸² Vo, B.-T. et al.: The Cardinality Balanced Multi-Target Multi-Bernoulli Filter and Its Implementations (2009).

⁸³ Hu, B. et al.: Random Finite Set Based Bayesian Filtering with OpenCL in a Heterogeneous Platform (2017).

⁸⁴ Reuter, S. et al.: Real-Time Multi-Object Tracking using Random Finite Sets (2013).

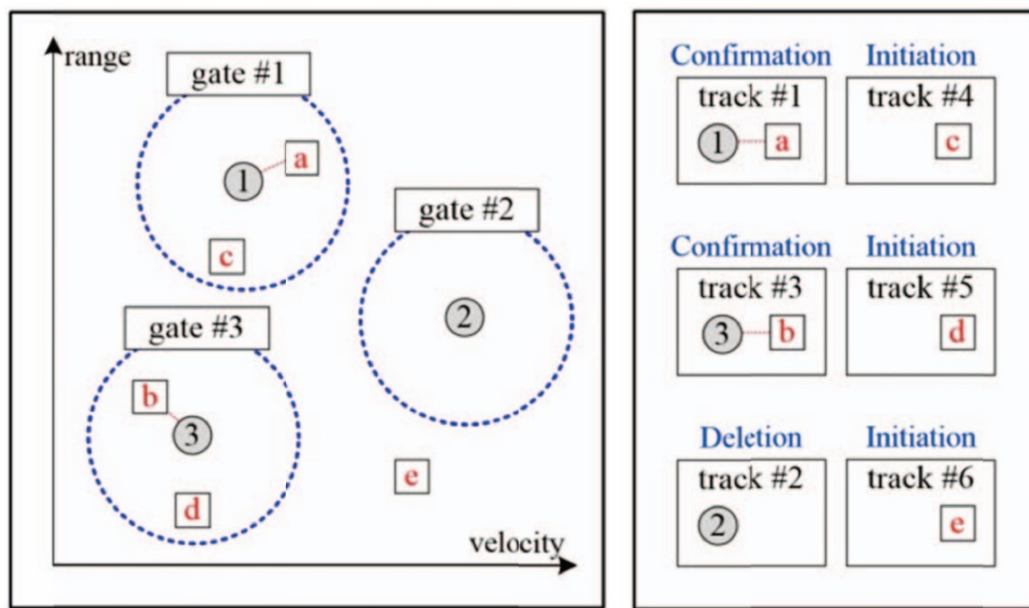


Abbildung 4-6: Mögliche Zustandsänderung von verfolgten Objekten^{85a}

Es gibt zwei Ansätze zur Umsetzung des Track-Management. Der erste Ansatz betrachtet dabei die Zahl der erfolgten oder nicht erfolgten Datenassoziationen vorheriger Messzyklen und wird als History-based bezeichnet. Eine Methode dieses Ansatzes ist das m/N-Verfahren. Ein Objekt wird als bestätigt angesehen, wenn in den letzten N Messungen insgesamt m Datenassoziationen erfolgten.⁸⁶ Hyun et al.^{85b} stellt eine weitere History-based Methode vor, welcher das Track-Management auf Basis eines Zustandsautomaten umsetzt. Im Gegensatz zum m/N-Verfahren wird hier ein Objekt bestätigt, wenn eine Datenassoziation in jedem konsekutiven Zyklus der Initiierungsphase erfolgte. Die Bedingungen zur Objektbestätigung sind demnach strenger als in der m/N-Methode.

Der zweite Ansatz zum Track-Management ist eine probabilistische Methode und unter dem Namen Score-based oder Sequential Probability Ratio Test bekannt. Zur Bestätigung und Löschung der Objekte wird die Wahrscheinlichkeit, dass das Objekt real existiert, aus der Detektionswahrscheinlichkeit, der Wahrscheinlichkeit eines Störsignals und weiteren probabilistischen Kennzahlen ermittelt.⁸⁶ Aus dieser Wahrscheinlichkeit ergibt sich der logarithmische Score eines Objekts. Mit dem Vergleich zu zuvor definierten Schwellenwerten erfolgt dann die Zustandsänderung. Eine Implementierung dieser Methode wird von Li et al.⁸⁷ vorgestellt.

Aufgrund der Kompatibilität des GNN-Verfahrens zur Datenassoziation mit History-based Ansätzen, da beide Methoden nicht auf probabilistischen Ansätzen basieren, wird für den zu implementierenden Algorithmus das m/N-Verfahren zum Track-Management genutzt. Um die Initiierung von Objekten bei bereits einer fehlenden Datenassoziation nicht abubrechen und so ein instabileres Tracking-Ergebnis zu erhalten, wird auf die Umsetzung des Zustandsautomaten nach Hyun et al. verzichtet.

⁸⁵ Hyun, E.; Lee, J.-H.: Track Management Table, a: S. 2; b: -.

⁸⁶ Blanding, W. R. et al.: Multisensor Track Management for Targets with Fluctuating SNR (2009), S. 1275.

⁸⁷ Li, X. R. et al.: SPRT-Based track confirmation and rejection (2002).

4.7 Prädiktion und Innovation

Im Schritt der Prädiktion erfolgt die Vorhersage der Objektzustände, des aktuellen Zyklus für den nächsten Zyklus mithilfe des zuvor definierten Bewegungsmodells. Der prädizierte Zustand wird anschließend zur Datenassoziation genutzt und bei erfolgter Zuordnung von Messwerten mit diesen korrigiert. Diese Korrektur bzw. Filterung nutzt das zuvor definierte Sensormodell und wird als Innovation beschrieben. Je nach genutztem Filter und Bewegungsmodell erfolgt die Prädiktion und Innovation auf unterschiedliche Weise. Die folgenden Abschnitte beschreiben die verschiedenen Bewegungsmodelle und stellen anschließend unterschiedliche Filteransätze vor.

4.7.1 Bewegungsmodelle zur Prädiktion

Bewegungsmodelle können entweder eine linear oder nicht-lineare Objektbewegung annehmen. Ein Beispiel für die lineare Bewegung ist das sogenannte Constant Velocity Model. Nicht-lineare Objektbewegungen wie Beschleunigungen werden zur Prädiktion zwischen zwei Zyklen vernachlässigt. Der Zustandsvektor eines Objekts besteht in diesem Modell aus vier Parametern:

$$\mathbf{x} = [x \ y \ \dot{x} \ \dot{y}]^T \quad (4-7)$$

Die Objektposition im Fahrzeugkoordinatensystem wird durch x und y angegeben. Die Objektgeschwindigkeit durch \dot{x} und \dot{y} . Die Prädiktion für den Zeitpunkt k ergibt sich aus

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{v}_k. \quad (4-8)$$

Die Unsicherheiten der Bewegung, also das Modellrauschen, wird mit \mathbf{v}_k eingebracht. Die Zustandsübergangsmatrix \mathbf{F} beschreibt dabei die Zustandsänderung des Objekts. Im Constant Velocity Model lautet diese:

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (4-9)$$

wobei T die Zykluszeit ist.^{88a}

Ein weiteres Beispiel ist das Constant Acceleration Model. Hierbei handelt es sich um ein nicht-lineares Bewegungsmodell, da Objektbeschleunigungen berücksichtigt werden. Mit dem Zustandsvektor $\mathbf{x} = [x \ y \ \dot{x} \ \dot{y} \ \ddot{x} \ \ddot{y}]^T$, wobei \ddot{x} und \ddot{y} die Beschleunigungen darstellen, ergibt sich die Zustandsübergangsmatrix nach (4-8) zu:^{88b}

⁸⁸ Pietzsch, S.: Diss., Sensordatenfusion zur Erfassung komplexer Fahrzeugumgebungen (2015), a: S. 50–51; b: S. 51–52.

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & T & 0 & \frac{T^2}{2} & 0 \\ 0 & 1 & 0 & T & 0 & \frac{T^2}{2} \\ 0 & 0 & 1 & 0 & T & 0 \\ 0 & 0 & 0 & 1 & 0 & T \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (4-10)$$

Ein weiteres Bewegungsmodell ist das nicht-lineare Constant Turn Model, welches die Gierrate des Objekts zur Berechnung der Zustandsänderung nutzt. Die Gierrate wird als konstant angenommen. Der Zustandsvektor lautet damit $\mathbf{x} = [x \ y \ \dot{x} \ \dot{y} \ \dot{\phi}]^T$, wobei $\dot{\phi}$ die Gierrate des Objekts darstellt. Die Gleichungen zum Zustandsübergang werden an dieser Stelle nicht explizit erwähnt.⁸⁹

Alle genannten Bewegungsmodelle beschreiben jeweils eine isolierte Bewegung. Verkehrsteilnehmer führen in der Realität allerdings nur selten eine einzige Bewegungsart durch. Aufgrund dessen gibt es Ansätze, die mehrere Bewegungsmodelle miteinander vereinen wie das Interacting Multiple-Model (IMM). Dabei wird der Übergang zwischen den Bewegungsmodellen mit Markow-Ketten geregelt. Die Ergebnisse der einzelnen Bewegungsmodelle werden so gewichtet und zusammengefasst. Eine finale Zustandsprädiktion ergibt sich damit aus allen Bewegungsmodellen.⁹⁰ Kim und Hong zeigen die Implementierung eines IMM mit dem Unscented Kalman Filter.⁹¹ Neben dem Vorteil der realitätsnahen Zustandsprädiktion liegt der Nachteil dieser Methode in der erhöhten Komplexität durch die Nutzung von Übergangswahrscheinlichkeiten.

4.7.2 Filtermethoden

Obwohl die Bayes-Gleichung aus (4-4) sich rekursiv berechnen lässt, ist sie aufgrund der Integralfunktion nicht zur Implementierung geeignet. Um dennoch das Ergebnis der Bayes-Gleichung abzuschätzen, wurden unterschiedliche Annäherungsverfahren entwickelt, welche nachfolgend vorgestellt werden.⁹²

Kalman Filter

Eines der Annäherungsverfahren ist das sogenannte Kalman Filter, welches ein weit verbreiteter Filteransatz für Tracking-Probleme ist. Diese Methode wird bei linearen Bewegungen und gaußförmigen Rauschen eingesetzt. Die genutzte Messmatrix des Sensormodells, also der Zusammenhang zwischen Zustandsvektor und Sensordaten, ist ebenfalls linear. Das Kalman Filter nutzt das zuvor definierte Bewegungsmodell zur Zustandsprädiktion. Mit jedem Zyklus wird ebenfalls die Kovarianz jedes einzelnen Zustands aus dem Rauschen des Bewegungsmodells prädiziert und konsekutiv aktualisiert.

⁸⁹ Schuster, M.: Diss., Multiple Object Tracking using JIPDA (2017), S. 196.

⁹⁰ P. Blom, H.: An efficient filter for abruptly changing systems (1984 - 1984).

⁹¹ Kim, Y.-S.; Hong, K.-S.: An IMM algorithm with federated information mode-matched filters for AGV (2007).

⁹² Liu, F.: Diss., Objektverfolgung durch Sensordatenfusion (2009), S. 33.

Im Korrekturschritt des prädizierten Objektzustands wird aus der jeweiligen Kovarianz und dem Rauschen des Sensormodells der sogenannte Kalman Gain ermittelt. Dieser Faktor bestimmt den Einfluss der Messdaten auf den prädizierten Objektzustand. Je größer dieser Einfluss ist, desto größer ist der Einfluss der Messdaten und die Dynamik des Systems steigt. Sinkt der Faktor, ist das System stabiler gegenüber Ausreißer in den Messdaten.⁹³ Ist das genutzte Bewegungs- oder Sensormodell nicht linear, ist diese Methode nicht geeignet.

Um das Problem des Kalman Filters für nicht-lineare Anwendungen zu umgehen, wurden mehrere Erweiterungen entwickelt. Die Bekanntesten sind hier das Extended Kalman Filter (EKF), das Unscented Kalman Filter (UKF) sowie das Partikel-Filter (PF)

Extended Kalman Filter

Das EKF ist eine direkte Erweiterung des Kalman-Filters, da Beide die gleiche Struktur besitzen. Zusätzlich wird Rauschen hier ebenfalls als gaußförmig angenommen. Der Einsatz eines EKF bei nicht-linearen Problemen wird durch die Linearisierung des Bewegungs- und Sensormodells mithilfe einer Taylor-Approximation erster Ordnung ermöglicht. Die Zustandsübergangs- und Messmatrix ergeben sich aus den Jacobi-Matrizen der zuvor definierten nicht-linearen Zustandsübergangs- und Messfunktion $f(\mathbf{S}_{k-1})$ und $h(\mathbf{S}_k)$:

$$\begin{aligned} \mathbf{F}_k &= \left[\frac{\partial f(\mathbf{S}_{k-1})}{\partial (\mathbf{S}_{k-1})} \right]_{\bar{\mathbf{S}}_{k-1|k-1}}, \\ \mathbf{H}_k &= \left[\frac{\partial h(\mathbf{S}_k)}{\partial (\mathbf{S}_k)} \right]_{\bar{\mathbf{S}}_{k|k-1}} \end{aligned} \quad (4-11)$$

$\bar{\mathbf{S}}_{k-1|k-1}$ bzw. $\bar{\mathbf{S}}_{k|k-1}$ stellen die Arbeitspunkte, die Zustandsschätzung des vorherigen Zyklus, der Linearisierung dar. Die restlichen Schritte des EKF sind analog zum linearen Kalman-Filter. Das EKF eignet sich besonders für Probleme deren Nicht-Linearität nicht hoch ist. Dies muss je nach Problem individuell entschieden werden.⁹⁴

Unscented Kalman Filter

Das Unscented Kalman Filter kann ebenfalls für nicht-lineare Probleme verwendet werden. Allerdings erfolgt hier keine Linearisierung der Modelle um einen Arbeitspunkt. Die Verteilung der vorherigen Zustandsschätzung wird durch Zustandsproben, den sogenannten Sigma-Punkten, approximiert, um eine genauere Prädiktion als das EKF zu erzielen.⁹⁵

Die Zahl N der benötigten Sigma-Punkte ergibt sich aus der Dimension n der Zustände zu $N = 2n + 1$. Die Berechnung der Sigma-Punkte erfolgt aus den vorherigen Zustandsschätzungen und deren Kovarianzen. Die Punkte werden anschließend gewichtet und durch das nicht-lineare Bewegungsmodell transformiert. Die transformierten Punkte ergeben eine neue abgeschätzte

⁹³ Thrun, S. et al.: Probabilistic robotics (2010), S. 43.

⁹⁴ Liu, F.: Diss., Objektverfolgung durch Sensordatenfusion (2009), S. 35–36.

⁹⁵ Julier, S. J.; Uhlmann, J. K.: New extension of the Kalman filter to nonlinear systems (1997), S. 186.

Zustandsverteilung, aus welcher der prädizierte Zustand ermittelt wird. Das UKF vergleicht mithilfe der Messfunktion des Sensormodells die zu erwartenden Messgrößen aus der Zustandsprädiktionen mit den tatsächlichen Sensordaten. Hieraus folgt die Korrektur der prädizierten Zustände und deren Kovarianzen.⁹⁶

Partikel-Filter

Eine weitere Filtermethode ist das Partikel-Filter (PF). Dieses Verfahren basiert auf der numerischen Berechnung, des Integrals aus (4-4). Dabei werden, ähnlich dem UKF, die Approximation durch ausgewählte Zustandsproben erreicht, den sogenannten Partikeln. Das PF verarbeitet diese mithilfe der Monte-Carlo-Methode. Der Unterschied des PFs zum UKF liegt in der Anzahl und Auswahl der Partikel. Während das UKF nur eine Approximation zweiter Ordnung erzielt, approximiert das PF die Nichtlinearität vollständig durch eine sehr große Anzahl von Partikeln. Zusätzlich wird hier beliebig geformtes Rauschen angenommen. Daher stellt das PF zwar eine optimale Lösung zur Zustandsschätzung dar, erfordert allerdings eine hohe Rechenleistung aufgrund der großen Zahl an Partikeln.^{97a}

Probability Hypothesis Density Filter

Erfolgt die Lösung des Tracking-Problems durch den Einsatz von Random Finite Sets, werden Probability Hypothesis Density Filter verwendet, wie in Kapitel 4.5.4 erstmals erwähnt wurde. Das PHD-Filter basiert auf der rekursiven Berechnung der Wahrscheinlichkeitsdichten der verfolgten Objekte. Die Wahrscheinlichkeiten werden demnach nicht für jeden Zyklus neu berechnet, sondern konsekutiv aktualisiert. Das Verfahren stellt damit eine recheneffiziente Methode dar. Allerdings ist diese Methode nur effektiv in Szenarien mit hoher Detektionswahrscheinlichkeit.^{97b} Aufgrund dessen wurde das PHD-Filter von Mahler zum Cardinalized-PHD-Filter (CPHD) weiterentwickelt, welches die Detektionswahrscheinlichkeit berücksichtigt.⁹⁸ Erweiterungen des PHD-Filters stellen unter anderem Granström et al. als Gaussian-Mixture-PHD-Filter vor. Die Autoren zeigen eine Methode, welche nicht das gesamte Set an Messdaten zur Zustandsschätzung nutzt, sondern mit der Betrachtung ausgewählter Subsets eine effektive Abschätzung der Zustände erfolgt.⁹⁹ Eine weitere Abwandlung des PHD-Filters liefert Lundquist et al. Hier wird die Umsetzung eines CPHD-Filters als Gamma-Gaussian-Inverse-Wishart-Implementierung gezeigt. Diese Erweiterung erzielt genauere Ergebnisse als ein PHD-Filter bezüglich der Objektzahlenschätzung und der Trennfähigkeit von nah beieinander liegenden Objekten.¹⁰⁰ Eine Alternative zum PHD-Filter stellt das Multi-Bernoulli-Verfahren dar. Implementierungen von Beard et al. konnten im Vergleich zum CPHD-Filter genauere Ergebnisse in der Abschätzung der Objektzahl erzielen und ermöglichten eine höhere Kontinuität der Objektverfolgung.¹⁰¹

⁹⁶ Wan, E. A.; van der Merwe, R.: The unscented Kalman filter for nonlinear estimation (2000), S. 155.

⁹⁷ Schuster, M.: Diss., Multiple Object Tracking using JIPDA (2017), a: S. 12; b: S.19-20.

⁹⁸ Mahler, R.: PHD filters of higher order in target number (2007).

⁹⁹ Granstrom, K. et al.: Extended Target Tracking using a Gaussian-Mixture PHD Filter (2012).

¹⁰⁰ Lundquist, C. et al.: An Extended CPHD Filter and GGIW Implementation (2013).

¹⁰¹ Beard, M. et al.: Multiple Extended Target Tracking With Labeled Random Finite Sets (2016).

Mit den genannten Verfahren zur Zustandsprädiktion und -innovation wurde eine Übersicht zu den geläufigen Filtermethoden gegeben. Da die Messwerte des Sensors und die Größen der Objektzustände im Rahmen dieser Arbeit keinen linearen Zusammenhang haben, wird ein Filter implementiert, welcher ein nicht-lineares Sensormodell verarbeitet wie beispielsweise das EKF, UKF oder das Partikel-Filter. Aufgrund der erforderlichen hohen Rechenleistung des PFs wird auf die Umsetzung dieser Methode verzichtet. Mit der Zykluszeit des vorliegenden Sensors ist die Nicht-Linearität des Tracking-Problems bezüglich der Objektbewegung relativ gering. Im Extremfall einer hohen relativen Objektbeschleunigung in von $a_{\text{rel},x} = 5 \frac{\text{m}}{\text{s}^2}$, beträgt die Differenz in der Positionsschätzung zu einer angenommenen konstanten Geschwindigkeit zwischen zwei Zyklen bei $\Delta x = a_{\text{rel},x} \cdot \frac{T^2}{2} = 0,013 \text{ m}$. Aus diesem Grund wird für die Implementierung des Filters ein Extended Kalman Filter gewählt.

4.8 Zusammenfassung

Mit den theoretischen Grundlagen und den genannten Verfahren zur Implementierung wurde in diesem Kapitel eine umfassende Übersicht über gängige Methoden zum Lösen des Tracking-Problems geliefert. Für den zu entwickelnden Algorithmus wird das Global Nearest Neighbor Verfahren für den Schritt der Datenassoziation aufgrund seiner geringen Komplexität und hoher Recheneffizienz gewählt. Das Track-Management erfolgt auf Basis des m/N-Verfahrens. Zusätzlich erfolgt die Implementierung eines Extended Kalman Filters, um das nicht-lineare Sensormodell umzusetzen. Die zu implementierenden Methoden wurden nach den Kriterien der Komplexität, Implementierbarkeit innerhalb dieser Arbeit, Recheneffizienz und der erreichbaren Leistungsverbesserung ausgewählt. Der weitere Verlauf dieser Arbeit erläutert die Schritte in der Entwicklung des Algorithmus bzw. in der Implementierung der genannten Methoden und stellt diese anschaulich dar.

5 Systemarchitektur

Im folgenden Kapitel wird die Systemarchitektur des entwickelten Algorithmus dargestellt. Dabei werden auf alle benötigten Schritte zur Verarbeitung der Eingangsdaten bis hin zur finalen Objektliste und deren Ein- und Ausgabedaten eingegangen.

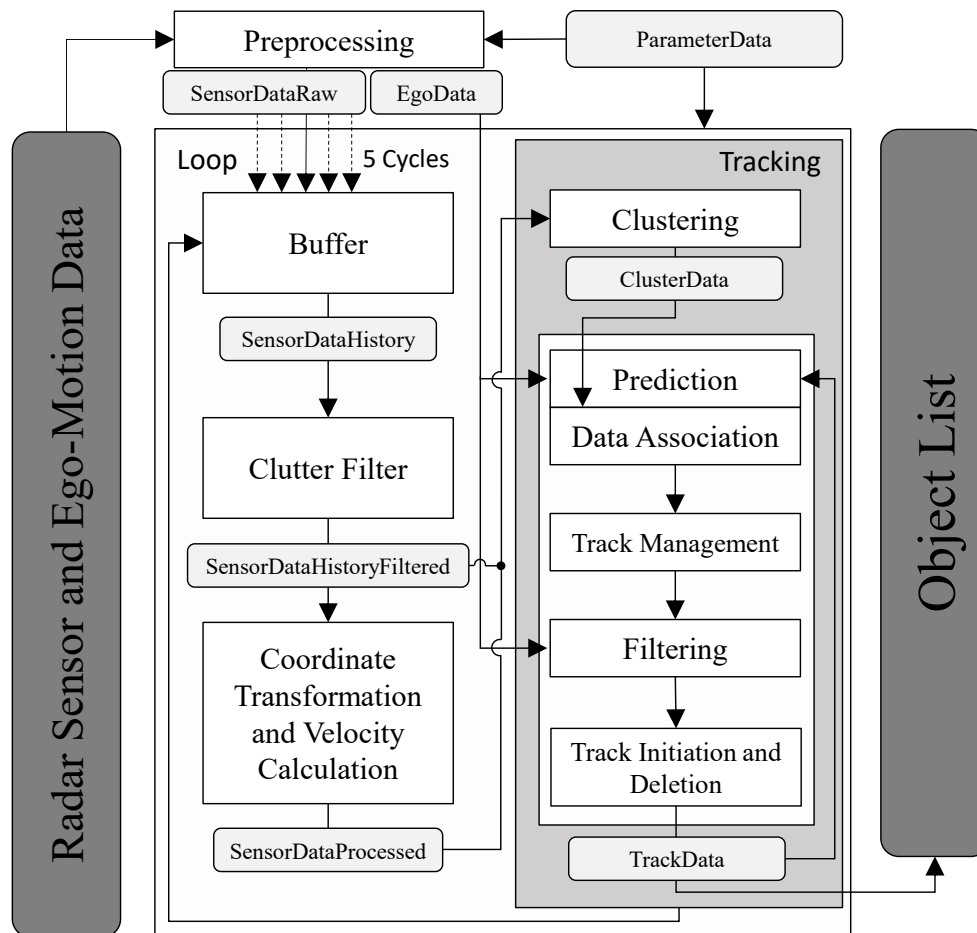


Abbildung 5-1: Systemarchitektur des entwickelten Algorithmus

Abbildung 5-1 zeigt die Systemarchitektur des gesamten Tracking-Algorithmus. Die Struktur ist gültig für synthetische sowie reale Eingangsdaten. Deren strukturelle Verarbeitung erfolgt demnach analog. Abgerundete Rechtecke in der Abbildung stellen Daten und deren Variablenname dar. Blöcke mit spitzen Ecken sind Funktionen, welche diese Parameter verarbeiten. Auf der linken Seite sind die Eingangsdaten abgebildet, während auf der rechten Seite die Ausgangsdaten stehen. Alle Schritte mit Ausnahme der Datenvorverarbeitung liegen in einer Programmierschleife (engl.: Loop), welche für jeden Zyklus ausgeführt wird. Innerhalb dieser Schleife liegt der Tracking-Algorithmus. Hier erkennt man die allgemeine Systemarchitektur, welche in Kapitel 4.3 vorgestellt wurde. Alle konstanten Größen, wie z.B. Schwellenwerte, Sensorauflösung oder Filterparameter, sind in *ParameterData* gespeichert. Diese Variable wird durch die entsprechenden Zuweisungen in einer Parameterdatei (*initParameterData*) initiiert. Die gewählten Werte der Parameter sind in Anhang A dargestellt.

Als Eingangsdaten werden Radarsensordaten sowie die Kinematikdaten des Egofahrzeugs benötigt. Die Kinematikdaten werden bei realen Eingangsdaten mit Hilfe einer Hardware im Egofahrzeug erfasst, welche unter anderem Positions- und Geschwindigkeitsdaten auf Basis von GPS-Daten und Inertialsensoren ermittelt. Für synthetische Daten werden diese Größen direkt aus der Simulationssoftware abgegriffen. Benötigt werden die absolute Geschwindigkeit des Egofahrzeugs sowie dessen Gierrate. Der Radarsensor liefert als Eingangsgröße die Bin-Daten zur Entfernung, zum Azimutwinkel und zur radialen Relativgeschwindigkeit¹⁰² der Detektionen sowie die Leistung des empfangenen Signals.

Die Vorverarbeitung (engl.: *Preprocessing*) der Eingangsdaten liest zum einen die Dateien unter dem im Hauptskript angegebenen Ordnerpfad aus und passt zum anderen deren Struktur für die weitere Verarbeitung an. Dabei wird anhand des Dateiformats automatisiert erkannt, ob es sich bei den Eingangsdaten um Simulations- oder Messdaten handelt, sodass das Preprocessing entsprechend abläuft. Ist das Preprocessing abgeschlossen, werden die Rohdaten des Sensors sowie Egokinematikdaten der gesamten Simulation bzw. Messung ausgegeben. Die Zyklen werden sequentiell in der Schleife verarbeitet. Diese Ausgangsdaten sind in der Abbildung als die Variablen `SensorDataRaw` und `EgoData` angegeben.

Die Verarbeitung der Daten erfolgt aufgrund der Berechnung der Dopplergeschwindigkeit aus den Bin-Daten nicht für jeden Zyklus separat, sondern mithilfe der Daten von insgesamt fünf Zyklen. Diese fünf Zyklen bestehen aus den zwei vorherigen, den zwei kommenden Zyklen sowie dem aktuellen Zyklus. Im Puffer (engl.: *Buffer*) werden diese Daten zwischengespeichert und für die kommende Verarbeitung in der Variable `SensorDataHistory` an den Algorithmus innerhalb der Tracking-Schleife übergeben. Das Ergebnis jeder Tracking-Schleife hat demnach einen Verzug von zwei Zyklen, d.h. die Ergebnisse von beispielsweise Zyklus 14 liegen erst in Zyklus 16 vor. Da ein Zyklus eine Zeit von $T = 0,072$ s hat, verursacht das Puffern einen Verzug von $2T = 0,144$ s.

Die nächste Funktion `Clutter Filter` filtert die Detektionen hinsichtlich ihrer Empfangsleistung. Da diese Größe dem Abstandsgesetz für Strahlenausbreitung unterliegt, wird für die Filterung die abstandskompensierte Leistung errechnet und mit einem zuvor definierten Schwellenwert verglichen. Liegt der Leistungswert unter diesem Schwellenwert, wird diese Detektion als Störsignal, Reflexion der Straßenoberfläche oder Vegetation am Fahrbahnrand klassifiziert und in der weiteren Verarbeitung nicht mehr betrachtet. Die gefilterten Detektionen sind in `SensorDataHistory-Filtered` zu finden.

Anschließend erfolgt im Block `Coordinate Transformation and Velocity Calculation` die Koordinatentransformation in kartesische Koordinaten sowie die Errechnung der Dopplergeschwindigkeit. Da die polaren Größen der Entfernung und des Azimutwinkels in Bin-Daten vorliegen, werden diese zuerst mit der jeweiligen Auflösung eines Bins multipliziert. Mithilfe des Azimutwinkels und der radialen Entfernung werden nun die Entfernungen in x- und y-Richtung des

¹⁰² Radiale Relativgeschwindigkeit und Dopplergeschwindigkeit werden zum Zweck der Ausdruckskürzung innerhalb dieser Arbeit synonym verwendet.

kartesischen Fahrzeugkoordinatensystems berechnet. Zusätzlich wird die Dopplergeschwindigkeit ermittelt. Die kartesischen Koordinaten sowie die errechnete Dopplergeschwindigkeit sind in `SensorDataProcessed` gespeichert.

Das Clustering des Tracking-Algorithmus nutzt die kartesischen Größen aus `SensorDataProcessed` sowie die Dopplerbins des aktuellen Zyklus aus `SensorDataHistoryFiltered`. Die zusammengefassten Detektionen werden in der Variable `ClusterData` abgelegt. Ist der durchlaufene Zyklus nicht der erste bzw. wurden bereits Objekte verfolgt, wird auf Basis dieser Daten, welche in `TrackData` gespeichert sind, in der Funktion `Prediction and Data Association` die Prädiktion durchgeführt. Um die Bewegungen des Egofahrzeuges zwischen dem aktuellen und dem letzten Zyklus für die Prädiktion zu berücksichtigen, erfolgt hier eine Eigenbewegungskompensation mithilfe der Daten aus `EgoData`. Die Funktion ordnet anschließend die Cluster den prädizierten Zuständen zu. Man erkennt hier einen Unterschied zur vorgestellten Systemarchitektur in Kapitel 4.3. Im entwickelten Algorithmus sind die Schritte der Prädiktion und der Datenassoziation aus Programmiergründen in einer Funktion zusammengefasst, damit die Prädiktion der Zustände nicht bereits im vorherigen Zyklus erfolgt und im anschließenden Zyklus der Datenassoziation übergeben werden muss. Da sich hier nur der Zeitpunkt der Prädiktion, aber nicht die prädizierten Zustände unterscheiden, hat diese Abwandlung keinen Einfluss auf das Tracking-Ergebnis.

Die Ergebnisse der Datenassoziation werden im `Track Management` interpretiert und dementsprechend die Statusänderungen der Objekte durchgeführt. Die Funktion legt damit fest, welche Objekte in diesem Zyklus gelöscht werden und damit keine Korrektur des prädizierten Zustands benötigen. In Abhängigkeit des neuen Status erfolgt damit der Korrekturschritt des Filters in der Funktion `Filtering`.

Die darauffolgende Funktion `Track Initiation and Deletion` untersucht die Zustände der korrigierten Objekte auf Duplikate. Duplikate können durch Mehrfachreflexionen eines Objektes entstehen, welche im Clustering aufgrund des definierten Schwellenwerts nicht zusammengefasst wurden, durch den Korrekturschritt im Filter aber ähnlichere Zustände aufweisen. Die Funktion errechnet die Differenzen der Geschwindigkeiten und Positionen aller Objekte zueinander und vergleicht diese mit einem definierten Schwellenwert. Liegen diese unter dem Schwellenwert, werden die zugehörigen Objekte als Duplikate klassifiziert, von denen nur das älteste Objekt behalten wird. Zusätzlich erfolgt in dieser Funktion die Initiierung neuer Tracks aus allen Clustern, die keinem Objekt zugewiesen wurden.

Die finalen Zustände aller Objekte in diesem Zyklus sind in der Variable `TrackData` gespeichert und werden für jeden Zyklus in die Objektliste geschrieben. Nach der Verarbeitung aller Zyklen, liegt die finale Objektliste in der Variable `ObjectList`. Hier stellt jede Spalte einen Zyklus dar, während jede Zeile im Verlauf der Mess- oder Simulationsdaten zu genau einem Objekt gehört.

Die Objektliste enthält für ein verfolgtes Objekt

- den Status (Initiierungsphase, bestätigte Phase oder Löschphase),
- die Entfernung, jeweils in x- und y-Richtung des Fahrzeugkoordinatensystem,
- die relative Geschwindigkeit, jeweils in x- und y-Richtung des Fahrzeugkoordinatensystem,
- die Dopplergeschwindigkeit,
- den Gierwinkel,
- die geschätzten Breite und Länge sowie
- die Eckpunkte der geschätzten Bounding-Box.

Unsicherheiten wie die im Kapitel 4.1 erläuterten Zustands-, Existenz- oder Klassenunsicherheit werden in der Objektliste nicht ausgegeben, da diese mit dem implementierten GNN-Verfahren zur Datenassoziation nicht berechnet werden. Zusätzlich zur Objektliste besteht die Möglichkeit, die Visualisierung zu aktivieren, welche für jeden Zyklus die Ergebnisse des Tracking-Algorithmus darstellt. Eine beispielhafte Darstellung ist in Anhang B gegeben.

6 Implementierung

Nachfolgend werden die implementierten Methoden einzelner Module des Tracking-Algorithmus genannt und erläutert. Die Reihenfolge orientiert sich dabei an der in Kapitel 5 vorgestellten Systemarchitektur. Es wird zu Beginn auf die Methode zur Filterung der Detektionen (6.1) eingegangen. Anschließend erfolgt die Darstellung zur Umrechnung der Bin-Daten in polare und kartesische Koordinaten (6.2). Die Erläuterung der implementierten Methoden im Clustering (6.3), in der Datenassoziation (6.4) und das Umsetzen des Track-Management (6.5) sind in den darauffolgenden Abschnitten zu finden. Eine umfassende Beschreibung der implementierten Verfahren zur Prädiktion und Filterung (6.6) folgt im Anschluss. Das Kapitel schließt mit der Darstellung der Methode zur Löschung von Duplikaten (6.7) ab.

Die Implementierung des Algorithmus wird in MathWorks MATLAB R2018b erstellt und nutzt vereinzelt zur Verfügung gestellte Programmcodes. Diese finden sich wieder

- in der Datenvorverarbeitung: `natsortfiles.m`¹⁰³ und `readNPY.m`¹⁰⁴
- im Clustering: `Clustering.m`¹⁰⁵
- im Extendenden Kalman Filter: `EKFprediction.m` und `EKFcorrection.m`¹⁰⁶
- in der Bestimmung der Kovarianzen im EKF: `GOSPA.m`¹⁰⁷

GOSPA steht hier für Generalized Optimal Sub-Pattern Assignment, welches eine Metrik zur Bewertung von Multi-Object-Tracking-Algorithmen darstellt. Diese Metrik betrachtet neben der Genauigkeit der Zustandsschätzung der verfolgten Objekte auch die Schätzung der Objektzahl.

6.1 Filterung der Detektionen

Zur Filterung der Detektionen von u.a. Störsignalen, Straßenreflexionen oder Reflexionen der Fahrbahnrandbepflanzung wird ein Schwellenwert für die empfangene Signalleistung definiert. Das Vorgehen zur Ermittlung dieses Wertes ist nachfolgend beschrieben. Das Szenario, welches für die Messdaten genutzt wurde, ist in Abbildung 6-1 dargestellt.¹⁰⁸ Das Objekt T_{t_0} stellt das Zielfahrzeug zum Startzeitpunkt t_0 dar. Die Anfangsposition liegt bei $r_0 = 18$ m und $\alpha_0 = 0$. Dieses Fahrzeug beschleunigt von $v_{\text{rel},t_0} = 0$ auf etwa $v_{\text{rel},t_1} = 30 \frac{\text{m}}{\text{s}}$ und durchfährt dabei den gesamten Sichtbereich des Sensors in x-Richtung.

¹⁰³ Cobeldick, S.: Natural-Order Filename Sort (2019).

¹⁰⁴ Kwik Team: NPY-Matlab (2018).

¹⁰⁵ Stefan Ackermann: Masterthesis, Systematische Untersuchung von Radar Tracking-Algorithmen (2017).

¹⁰⁶ Cao, Y.: Learning the Extended Kalman Filter (2008).

¹⁰⁷ Rahmathullah, A. S. et al.: GOSPA (2018).

¹⁰⁸ Dieses Szenario wurde mit einem BMW i3 ausgeführt. Objekte mit anderer Form reflektieren andere Signalleistungen. Der definierte Schwellenwert garantiert daher keine effektive Filterung für alle Objektarten.

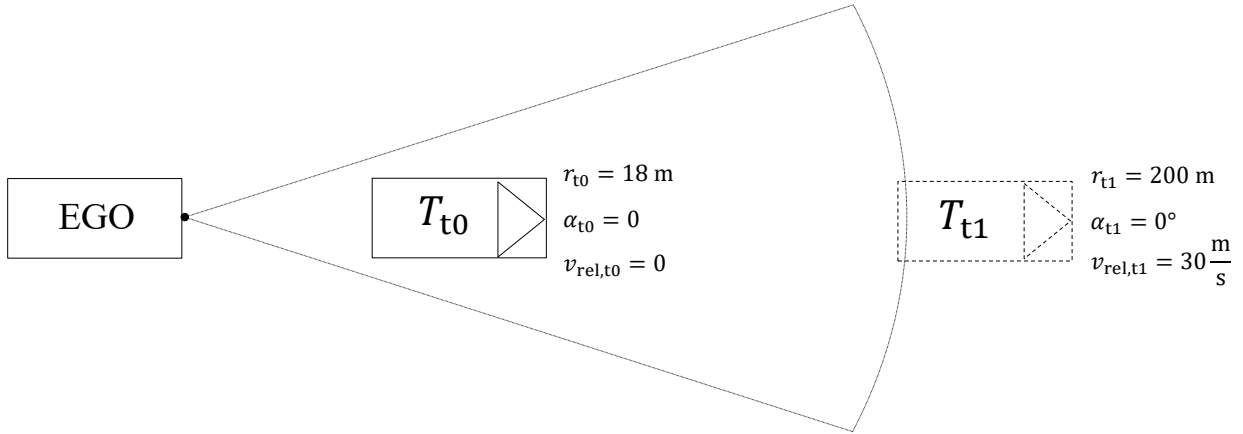


Abbildung 6-1: Szenario zur Messung der Signalleistung

Mit den erfassten Detektionen werden so die Unterschiede in den Signalleistungen irrelevanter Detektionen und Detektionen von Objekten für den gesamten Entfernungsbereich des Sensors ermittelt. Die empfangene Signalleistung ist eine Verstärkungsangabe und wird daher logarithmiert und in Dezibel angegeben. Die Größe unterliegt zudem dem Abstandsgesetz und ist damit antiproportional zur Entfernung der Detektion. Dieser Effekt ist in Abbildung 6-2 im oberen Diagramm zu sehen. Mit steigender Entfernung sinkt die reflektierte Leistung des Objekts. Da das Zielfahrzeug in diesem Szenario das einzige dynamische Objekt im Sichtbereich des Sensors war, lassen sich dessen Detektionen anhand der Dopplergeschwindigkeit eindeutig von den restlichen Detektionen unterscheiden. Auf den ersten Blick bietet sich hier ein Schwellenwert von $P_{\log} = -40 \text{ dB}$ an, welcher als rote Linie im oberen Diagramm eingezeichnet ist. Betrachtet man allerdings die Wertebereiche der Signalleistung des Objekts, liegt dieser bei etwa $[-50, -10]$, der Wertebereich der Störsignale bei etwa $[-75, -40]$. Da sich die Wertebereiche überschneiden, lässt für die reine Signalleistung kein effektiver Schwellenwert definieren. Die Größe wird daher mit folgender Formel abstandskompensiert

$$P_{\text{comp},\log} = 10 \log_{10}(P \cdot r^3) \quad (6-1)$$

$$P_{\text{comp},\log} = P_{\log} + 10 \log_{10}(r^3) \quad (6-2)$$

P ist die reine Signalleistung vor der Logarithmierung, r die jeweilige Entfernung der Detektion und $P_{\text{comp},\log}$ beschreibt die abstandskompensierte Leistung. Mithilfe der Logarithmengesetze lässt sich (6-1) umformen zu (6-2), sodass der bereits vorhandene Wert der logarithmischen Signalleistung in dB verwendet werden kann. Die so berechnete abstandskompensierte Leistung ist in Abbildung 6-2 im unteren Diagramm abgebildet.

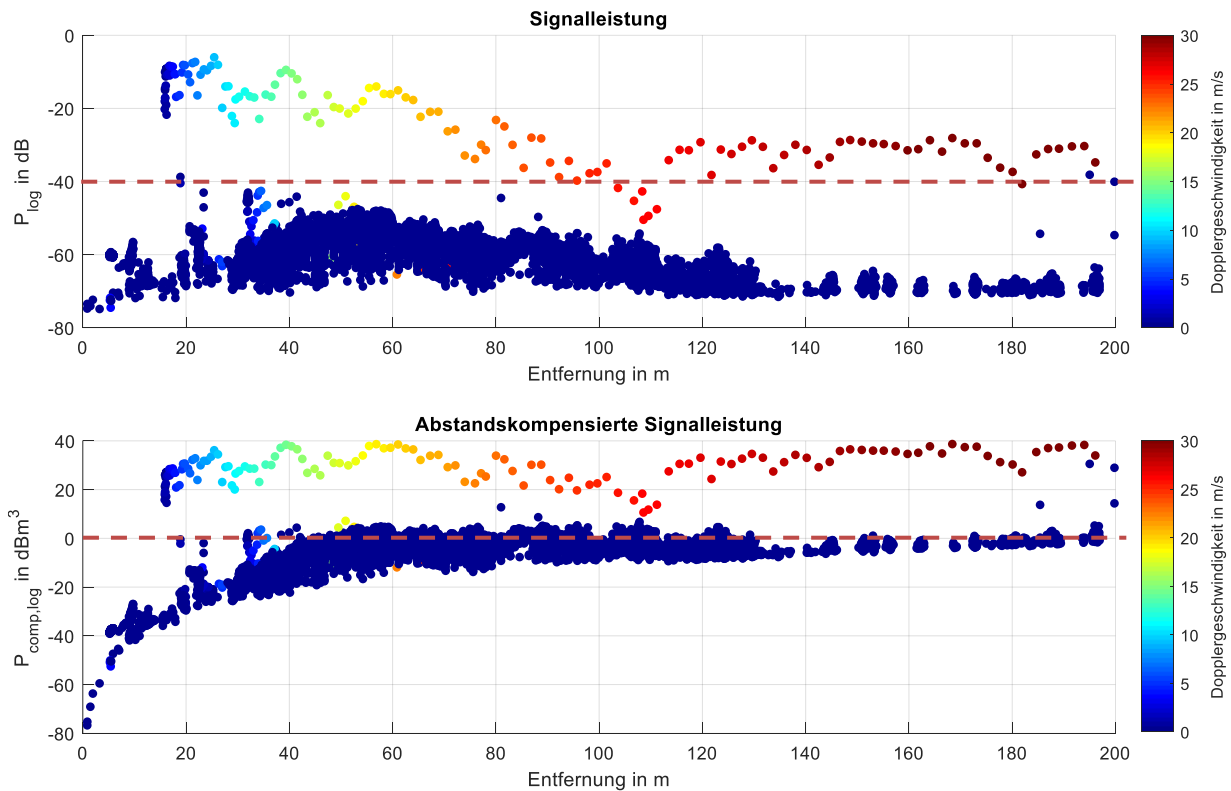


Abbildung 6-2: Vergleich der abstandskompensierten Signalleistung zur reinen Signalleistung über die Entfernung

Man erkennt hier im Vergleich zur Signalleistung im oberen Diagramm keinen stetigen Abfall der Leistung mit steigender Entfernung, sodass die Überschneidung der genannten Wertebereiche kleiner ausfällt. Ein Großteil der abstandskompensierten Signalleistungen der Störsignale liegt unter $P_{\text{comp,log}} = 0$. Obwohl einige Werte von Störsignalen über diesem Wert liegen, wird $P_{\text{comp,log}} = 0$ als Schwellenwert gewählt, um einen Kompromiss zwischen der Filterung von Störsignalen und der Gewährleistung der Verarbeitung von Objektdetektionen zu erreichen. Radarsensordaten, deren abstandskompensierte Signalleistung unter $P_{\text{comp,log}} = 0$ liegt werden damit im weiteren Verlauf der Algorithmus nicht mehr betrachtet.

6.2 Umrechnung der Bin-Daten in radiale und kartesische Koordinaten

Nachfolgend wird die Umrechnung der aus dem Radarsensor enthaltenen Bin-Daten in radiale und kartesische Koordinaten erläutert.

Entfernungs- und Azimutwinkelbin

Die Entfernungsberechnung in radiale Koordinaten stellt die simpelste Umrechnung der Bin-Daten dar, da zum einen die Bin-Daten keine Mehrdeutigkeiten aufweisen und zum anderen der Wertebereich der Entfernung ausschließlich positiv ist. Um die radiale Entfernung r zu berechnen, wird der Entfernungsbins r_{bin} mit der Auflösung der Entfernung Δr multipliziert:

$$r = r_{\text{bin}} \cdot \Delta r \quad (6-3)$$

Aus dieser radialen Entfernung lassen sich die kartesischen Entfernungen im Fahrzeugkoordinatensystem ermitteln. Dazu wird der eindeutige Azimutwinkel α benötigt, welcher aus der bekannten Auflösung $\Delta\alpha$ des Azimutwinkelbins α_{bin} errechnet wird. Da die nachfolgende Formel die Eindeutigkeit des Azimutwinkels voraussetzt, ist diese nur innerhalb des Eindeutigkeitsintervalls gültig, und lautet analog zu (6-3):

$$\alpha = \alpha_{\text{bin}} \cdot \Delta\alpha \quad (6-4)$$

Die kartesischen Entfernungen x und y ergeben sich dann aus

$$x = r \cdot \cos(\alpha) \quad (6-5)$$

$$y = r \cdot \sin(\alpha) \quad (6-6)$$

Dopplerbin

Zur Auflösung der Mehrdeutigkeiten im Dopplerbin stehen mehrere Verfahren wie beispielsweise der chinesische Restwertsatz zur Verfügung. Grundsätzlich ist auch hier das Vorgehen

$$\dot{r} = \dot{r}_{\text{bin}} \cdot \Delta\dot{r}, \quad (6-7)$$

bei dem \dot{r} die Dopplergeschwindigkeit, \dot{r}_{bin} den Dopplerbin und $\Delta\dot{r}$ die Auflösung des Dopplerbins darstellen.

6.3 Clustering

Das Clustering zur Zusammenfassung von mehreren Detektionen eines Objekts ist in zwei Phasen unterteilt und erfolgt auf Basis der Entfernungen im kartesischen Fahrzeugkoordinatensystem und den zugehörigen Dopplerbins. Die erste Phase teilt die Detektionen des Zyklus anhand von Ähnlichkeiten in der Position und der Dopplerbins in Cluster auf. Die zweite Phase ergänzt Geschwindigkeiten für Detektionen, denen in der ersten Berechnung der Dopplergeschwindigkeit kein Wert eindeutig zugewiesen werden konnte. Diese Ergänzung erfolgt durch den Vergleich der potenziellen Dopplergeschwindigkeit des Dopplerbins mit bestätigten Geschwindigkeiten der restlichen Detektionen des entsprechenden Clusters. Mithilfe dieser Methode wird die Betrachtung von Detektionen, welche ansonsten durch die alleinige Berechnung der Geschwindigkeiten verworfen worden wären, ermöglicht. Die detaillierte Vorgehensweise beider Methoden ist nachfolgend beschrieben.

Die erste Phase berechnet für einen Zyklus die euklidischen Distanzen sowie die Unterschiede der Dopplerbins aller Detektionen zueinander. Diese Größen liegen in einer symmetrischen Matrix, dessen Spalten- bzw. Zeilenzahl sich aus der Zahl der vorliegenden Detektionen ergibt. Die errechneten Werte der Matrix werden anschließend mit jeweils definierten Schwellenwerten verglichen. Diese sind innerhalb dieser Arbeit für die euklidische Distanz bei $d_{\text{max}} = 5 \text{ m}$ sowie bei $\Delta\dot{r}_{\text{bin,max}} = 20$ für die Differenz der Dopplerbins festgelegt. Liegen die euklidische Distanz und die Differenz der Dopplerbins zweier Detektionen unter diesen Schwellenwerten, werden diese Detektionen als zusammenhängend angenommen. Eine anschließende rekursive Funktion gruppiert alle zusammenhängenden Detektionen zu einzelnen Clustern. Dabei müssen in einem Cluster nicht alle Detektionen zueinander zusammenhängend sein, sondern der Kontakt einer Detektion zu mindestens einer weiteren reicht

hier aus. Abbildung 6-3 zeigt ein Beispiel dafür. Die euklidischen Distanzen sind auf den Pfeilen zwischen den Detektionen 1 bis 4 in Metern vermerkt. Die Dopplerbins der Detektionen finden sich auf der rechten Seite.

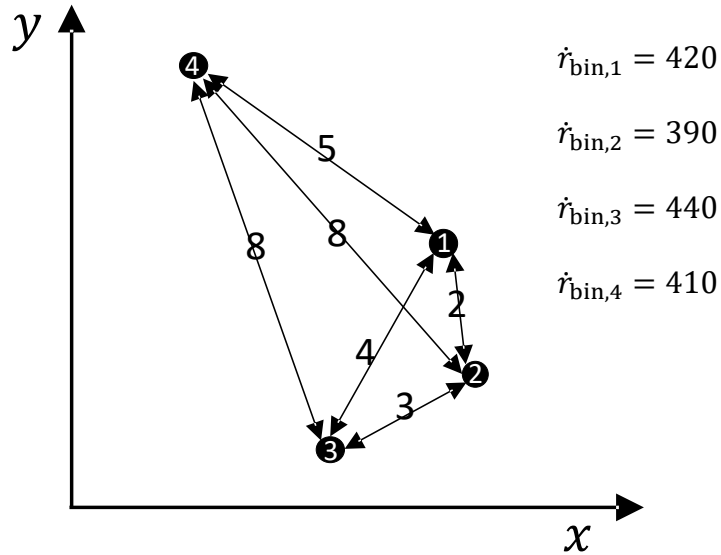


Abbildung 6-3: Euklidische Distanzen zwischen Detektionen und deren Dopplerbins

Die erwähnten symmetrischen Matrizen, welche die Entfernung \underline{d} und die Differenzen der Dopplerbins zueinander $\underline{\Delta \dot{r}_{\text{bin}}}$ enthalten, lauten für dieses Beispiel wie folgt:

$$\underline{d} = \begin{bmatrix} 0 & 2 & 4 & 5 \\ 2 & 0 & 3 & 8 \\ 4 & 3 & 0 & 8 \\ 5 & 8 & 8 & 0 \end{bmatrix}, \quad \underline{\Delta \dot{r}_{\text{bin}}} = \begin{bmatrix} 0 & 40 & 20 & 10 \\ 40 & 0 & 50 & 20 \\ 20 & 50 & 0 & 30 \\ 10 & 20 & 30 & 0 \end{bmatrix} \quad (6-8)$$

Dabei beschreibt der Eintrag in beispielsweise Zeile 2 und Spalte 3 die euklidische Distanz bzw. die Differenz der Dopplerbins der Detektionen mit der Nummer 2 und 3. Werden diese beiden Matrizen nun mit den Schwellenwerten $d_{\text{max}} = 5 \text{ m}$ und $\Delta \dot{r}_{\text{bin,max}} = 20$ verglichen, ergeben sich die binären Matrizen:

$$\underline{d}^{01} = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}, \quad \underline{\Delta \dot{r}_{\text{bin}}}^{01} = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix} \quad (6-9)$$

Eine 1 bedeutet dabei, dass der Wert der Matrix \underline{d} bzw. $\underline{\Delta \dot{r}_{\text{bin}}}$ an dieser Stelle unter dem jeweiligen Schwellenwert liegt. Da beide Bedingungen der euklidischen Distanz und der Differenz der Dopplerbins erfüllt sein müssen, folgt die Verknüpfung der binären Matrizen zu $\underline{d}^{01} \cap \underline{\Delta \dot{r}_{\text{bin}}}^{01}$:

$$\underline{d}^{01} \cap \underline{\Delta \dot{r}_{\text{bin}}}^{01} = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \quad (6-10)$$

Aus dieser Matrix lässt sich ableiten, dass die Detektion 1 mit den Detektionen 3 und 4 zusammenhängt, während die Detektionen 3 und 4 untereinander nicht zusammenhängend sind. Da der Kontakt zu nur einer Detektion in einem Cluster ausreicht, wird das erste Cluster durch die Detektionen 1, 3 und 4 gebildet., wie in Abbildung 6-4 dargestellt ist. Das zweite Cluster besteht nur aus Detektion 2. Die erste Phase des Clusterings ist damit abgeschlossen.

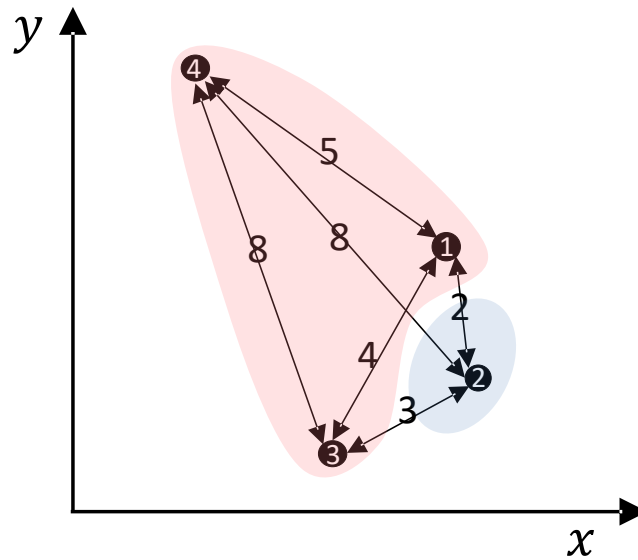


Abbildung 6-4: Finale Clusterzuordnung

Die zweite Phase des Clusterings betrachtet Detektionen, denen keine errechnete Dopplergeschwindigkeit zugeordnet werden konnte. Die Methode beruht auf der Annahme, dass mehrere Detektionen eines Objekts ähnliche Dopplergeschwindigkeiten besitzen und damit mit dem gleichen Wert für i_{hyp} berechnet werden können. Beispielsweise besitzt Detektion 3 keinen errechneten Dopplerwert für den zugehörigen Dopplerbin $\dot{r}_{bin,3} = 440$. Daher werden die hypothetischen Dopplergeschwindigkeiten dieser Detektion, also die Geschwindigkeiten aller Hypothesen, mit den eindeutig zugewiesenen Geschwindigkeiten der restlichen Detektionen des Clusters verglichen. Liegt die Differenz einer der hypothetischen Geschwindigkeit zu den errechneten Geschwindigkeiten unter dem Schwellenwert von $\Delta\dot{r}_{max} = 2 \frac{m}{s}$, wird diese hypothetische Geschwindigkeit der Detektion zugeordnet. Die Geschwindigkeiten aller Hypothesen für den Dopplerbin $\dot{r}_{bin,3} = 440$ und die errechneten Geschwindigkeiten \dot{r}_1 und \dot{r}_4 für die restlichen Detektionen des Clusters lauten wie folgt:

Tabelle 2: Potenzielle Dopplergeschwindigkeiten der Detektion 3 im Vergleich zu errechneten Dopplergeschwindigkeiten der restlichen Detektionen 1 und 4

Wertebereich	Hypothesennummer	\dot{r} für $\dot{r}_{bin,3} = 440$ in m/s	\dot{r}_1	\dot{r}_4
positiv	$i_{hyp} = 1$	7,22	-	-
	$i_{hyp} = 2$	58,58	60,58	61,58
negativ	$i_{hyp} = 1$	-44,13	-	-

$i_{\text{hyp}} = 2$	-95,49	-	-
$i_{\text{hyp}} = 3$	-146,84	-	-

Hier ist zu erkennen, dass die Geschwindigkeit zur Hypothesennummer $i_{\text{hyp}} = 2$ aus dem positiven Wertebereich die Bedingung zum Schwellenwert $\Delta \dot{r}_{\text{max}} = 2 \frac{\text{m}}{\text{s}}$ erfüllt. Damit erfolgt hier die Zuordnung der Dopplergeschwindigkeit $\dot{r}_3 = 58,58 \frac{\text{m}}{\text{s}}$ der Detektion 3.

Kann einer Detektion mit dieser Methode ebenfalls keine Geschwindigkeit zugeordnet werden, oder besteht ein Cluster nur aus Detektionen ohne zuvor errechneten Werten, werden diese verworfen und im weiteren Verlauf des Tracking-Algorithmus nicht betrachtet.

6.4 Datenassoziation

Der Schritt der Datenassoziation ordnet mit Betrachtung der Positionen und Dopplergeschwindigkeiten die gebildeten Cluster den prädizierten Objektzuständen. Die Differenz zwischen diesen Größen wird als Innovation bezeichnet.¹⁰⁹ Die Referenz der verglichenen Positionen des Clusters und des Objekts ist hierbei der geometrische Schwerpunkt der geschätzten Bounding-Box. In Kapitel 6.6.5 ist die Methode zur Schätzung einer Bounding-Box um einzelne Detektionen eines Clusters oder eines Objekts dargestellt. Der geometrische Schwerpunkt wird zur Datenassoziation dem arithmetischen Mittel der einzelnen Detektionspositionen vorgezogen, da die Verteilung der erfassten Detektionen um ein Objekt in der Regel nicht gleichförmig ist. Die Berechnung des arithmetischen Mittels liefert daher nicht den Mittelpunkt des Objekts. bzw. des Clusters.¹⁰⁹

Innerhalb eines Zyklus werden die Innovationen der Geschwindigkeit I_{dop} und der Position I_{pos} für jedes Cluster zu allen prädizierten Objektzuständen berechnet. Die Innovationen eines Clusters zu einem Objekt ergeben sich dabei aus

$$I_{\text{pos}} = (x_{\text{obj}} - x_{\text{clust}})^2 + (y_{\text{obj}} - y_{\text{clust}})^2 \quad (6-11)$$

$$I_{\text{dop}} = (\dot{r}_{\text{obj}} - \dot{r}_{\text{clust}})^2, \quad (6-12)$$

wobei x_{obj} und y_{obj} die Entfernungen des Objekts in x- und y-Richtung des Fahrzeugkoordinatensystem sind. Analog dazu sind die Entfernungen x_{clust} und y_{clust} des Clusters. Die Dopplergeschwindigkeiten des Objekts und des Clusters werden jeweils durch \dot{r}_{obj} und \dot{r}_{clust} beschrieben. Die Zuordnung eines Clusters zu einem Objekt erfolgt nun, wenn die jeweiligen Innovationen beide unter den zuvor definierten Schwellenwerten von $I_{\text{pos,max}} = 9 \text{ m}^2$ und $I_{\text{dop,max}} = 9 \frac{\text{m}^2}{\text{s}^2}$ liegen und die Summe der Innovationen zu diesem Objekt das Minimum aller erreichten Innovationen dieses Clusters ist. Liegen die Innovationen eines Clusters also zu mehreren Objekten unter diesen Schwellenwerten, wird das Cluster nur mit dem Objekt assoziiert, zu dem die Summe der Innovationen am geringsten ist. Mehrfachzuordnungen, wie sie in 4.5 vorgestellt wurden, sind damit ausgeschlossen. Einem

¹⁰⁹ Gosala, N. B.; Meng, X.: An RLS-Based Instantaneous Velocity Estimator for Extended Radar Tracking (2020), S. 2.

Objekt können allerdings mehrere Cluster zugeordnet werden. Dies ist beispielsweise im Fall von langen Objekten erforderlich. Die Definition der Schwellenwerte basiert auf der Arbeit von Gosala und Meng, welche die Implementierung eines Tracking-Algorithmus für Echtzeitanwendungen vorstellen.¹¹⁰

6.5 Track-Management

Das Track-Management des implementierten Algorithmus verwaltet die Status der Objekttracks. Je nach Phase kann ein Track vier verschiedene Status besitzen. Die Phasen lauten wie folgt:

- Inaktive Phase (Idle): Unter der jeweiligen Track-ID ist kein Objekt vorhanden.
- Initiierungsphase (Initiation): Ein neues Objekt wurde initiiert. In den folgenden Zyklen wird geprüft, ob es sich um ein reales Objekt oder Geisterobjekt handelt. Ein Geisterobjekt entsteht beispielsweise durch nicht gefilterte Störsignale oder Straßenreflexionen.
- Bestätigte Phase (Confirmation): Ein zuvor initiiertes Objekt wurde über eine bestimmte Zahl von Zyklen erfolgreich verfolgt und befindet sich daher nun in der bestätigten Phase.
- Löschphase (Deletion): Ein bestätigtes Objekt wurde verloren. Erfolgt für dieses Objekt in den weiteren Zyklen ebenfalls keine Datenassoziation, wird dieses Objekt gelöscht.

Der Übergang von der Initiierungs- in die bestätigte Phase erfolgt auf Basis des m/N -Verfahrens, welches in Kapitel 4.6 vorgestellt wurde. Dabei gilt ein Objekt als bestätigt, sobald innerhalb von N Zyklen mindestens m Datenassoziationen zu diesem Objekt erfolgten. Alle Status und deren Übergänge sind in Abbildung 6-5 dargestellt. Dabei stellen die steil dargestellten Parameter die Bedingung für diesen Übergang dar, während die kursiv geschriebenen Parameter eine Aktion beschreiben, wie zum Beispiel die Hochzählung eines bestimmten Parameters. Die Bedeutung der genutzten Parameter in der Abbildung und deren Wertebereiche bzw. Schwellenwerte für Statusübergänge sind in Tabelle 3 gelistet. Dabei ist der Parameter *InitCount* der Größe N sowie *DACount* der Größe m des m/N -Verfahrens gleichzusetzen.

Tabelle 3: Genutzte Variablen für Statusübergänge

Variable	Wertebereich	Schwellenwert	Bedeutung
DA	{0,1}	-	1 = Erfolgreiche Datenassoziation; 0 = Keine Datenassoziation mit Cluster
InitCount	{0,1,3,4,5}	5	Zähler für alle Zyklen in der Initiierungsphase
DACount	{0,1,2,3}	3	Zähler für Zyklen in der Initiierungsphase mit erfolgreicher Datenassoziation

¹¹⁰ Gosala, N. B.; Meng, X.: An RLS-Based Instantaneous Velocity Estimator for Extended Radar Tracking (2020), S. 3.

MissedDACount	{1, ...,10}	10	Zähler für konsekutive Zyklen ohne erfolgreiche Datenassoziation
---------------	-------------	----	--

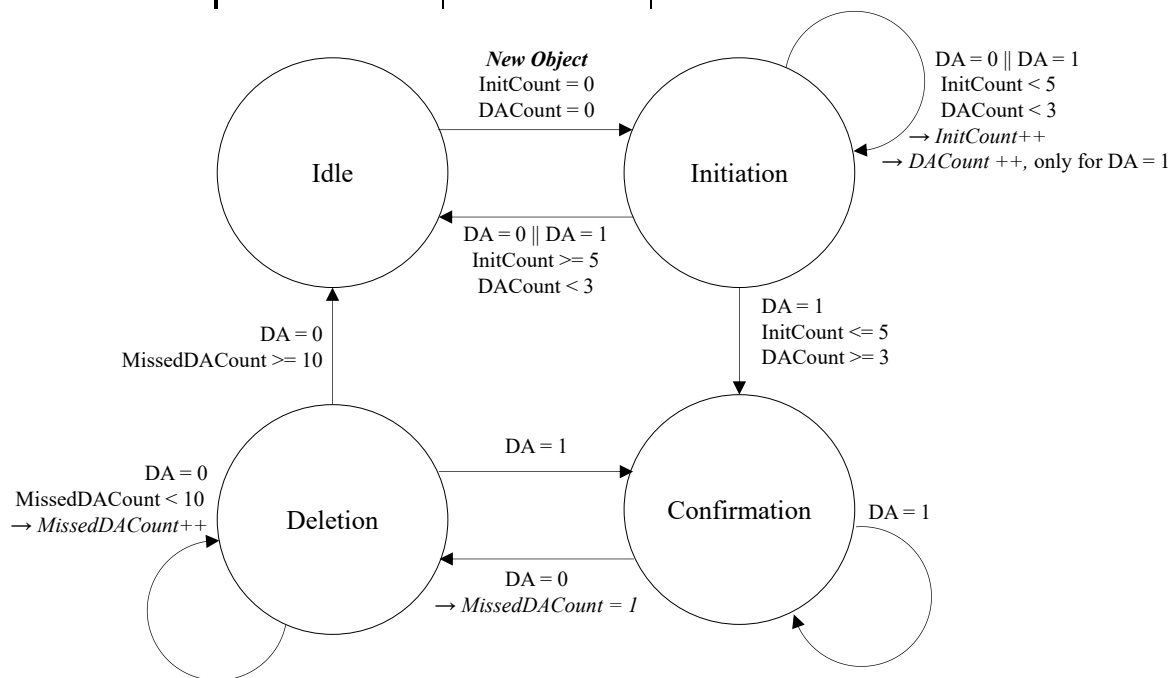


Abbildung 6-5: Statusübergänge der Objekttracks

Aus der Abbildung ist zu erkennen, dass, sobald ein neues Objekt aus einem nicht zugeordneten Cluster verfolgt wird, eine nicht belegte Track-ID aus der inaktiven Phase (Idle) in die Initiierungsphase übergeht. Die Parameter *InitCount* und *DACount* werden innerhalb dieser Phase genutzt, um den nächsten Status dieses Objekts zu entscheiden. Erfolgen in den kommenden fünf Zyklen drei Datenassoziationen, geht dieses Objekt in die bestätigte Phase über. Wird die Bedingung nicht erfüllt, war die Initiierung nicht erfolgreich und die zugehörige Track-ID geht erneut in die inaktive Phase über. Einen Sonderfall stellt ein Objekt dar, dessen kartesischen Geschwindigkeiten bei der Initiierung nicht bestimmt waren und daher mit der Zuordnung eines Clusters im darauffolgenden Zyklus bestimmt werden müssen. Erfolgt im ersten Zyklus der Initiierungsphase allerdings keine Datenassoziation wird das Objekt direkt verworfen, da die kartesischen Geschwindigkeiten nicht bestimmt werden können, welche für eine zweite Prädiktion benötigt werden. Die Methode der initialen Prädiktion ist in Kapitel 6.6.1 erläutert.

Befindet sich ein Objekt in der bestätigten Phase, bleibt dieses bestehen, solange erfolgreiche Datenassoziationen erfolgen. Geschieht dies nicht, geht das Objekt mit der ersten nicht erfolgten Datenassoziation in die Löschphase über.

Die Löschphase nutzt den Parameter *MissedDACount*. Dieser wird mit jeder konsekutiven nicht erfolgten Datenassoziation hochgezählt. Können in den nächsten 10 konsekutiven Zyklen diesem Objekt kein Cluster zugeordnet werden, wird es gelöscht und die Track-ID geht in die inaktive Phase über. Erfolgt allerdings eine Datenassoziation, wird das Objekt erneut bestätigt und der Zähler *MissedDACount* zurückgesetzt. Diese neue Bestätigung ist damit unabhängig von den vorherigen Status.

6.6 Prädiktion und Filterung

Für die erste Prädiktion und Filterung eines Objektzustände wird zwischen folgenden Fällen unterschieden:

1. Bei der Initiierung des Objekts ist nur die Dopplergeschwindigkeit bekannt.
2. Bei der Initiierung des Objekts sind die kartesischen Geschwindigkeiten bekannt.

Die kartesischen Geschwindigkeiten eines Objektes können bei der Initiierung bekannt sein, sofern mehrere Detektionen im entsprechenden Cluster vorhanden sind, welche die Anwendung der Instant-Velocity-Estimation erlauben. Kapitel 6.6.4.1 erläutert diese Methode. Das detaillierte Vorgehen ist an dieser Stelle für die Prädiktion nicht relevant, sodass nachfolgend nur auf die zwei genannten Fälle eingegangen wird.

Für Objekte des ersten Falls erfolgt die initiale Prädiktion und Filterung. Für Objekte des zweiten Falls werden die Prädiktion und Filterung mit dem Extended Kalman Filter durchgeführt. Beide Methoden sind jeweils in den nächsten Abschnitten erläutert.

6.6.1 Initiale Prädiktion und Filterung

Die Prädiktion eines Objektes, für das nur die Dopplergeschwindigkeit bekannt ist, kann nicht mit dem Extended Kalman Filter erfolgen. Dieses benötigt zur Prädiktion der Richtung kartesische Geschwindigkeiten. Da mit der Dopplergeschwindigkeit nur der radiale Anteil der relativen Geschwindigkeit bekannt ist, ist in diesem Prädiktionsvorgehen der geschätzte zurückgelegte Weg des Objekts mit einer größeren Unsicherheit behaftet als im EKF. Aus diesen Gründen erfolgt mit dieser Methode keine Prädiktion auf einen spezifischen Punkt, sondern auf einen definierten Suchbereich in Form eines Halbkreises. Mit dem Vorzeichen der Dopplergeschwindigkeit und dem Azimutwinkel lässt sich die Richtung des Objekts auf einen Bereich mit einem Öffnungswinkel von 180° begrenzen. In Abbildung 6-6 ist für zwei beispielhafte Objekte mit der zur relativen Geschwindigkeit $v_{i,\text{rel}}$ (dem Sensor unbekannt) gehörenden Dopplergeschwindigkeit \dot{r}_i und dem Azimutwinkel α_i dieser Suchbereich grau hinterlegt dargestellt. Die Sekante durch den Kreismittelpunkt zur Begrenzung des Halbkreises liegt stets orthogonal zur Richtung des Entfernungsvektors, sodass der Schnittwinkel mit der y-Achse gleich dem Azimutwinkel ist. Ist die Dopplergeschwindigkeit positiv, entspricht der Suchbereich der Kreishälfte, welche vom Koordinatenursprung weg zeigt. Bei negativer Dopplergeschwindigkeit wird die Kreishälfte genutzt, welche zum Koordinatenursprung zeigt. Der Radius dieses Halbkreises und damit die Begrenzung des Suchbereichs Δr_{pred} orientieren sich an der Dopplergeschwindigkeit. Da diese, wie in Abbildung 6-6 zu sehen ist, nur den radialen Anteil der relativen Geschwindigkeit beschreibt, wird sie mit dem Faktor 2,5 multipliziert, und ergibt sich aus

$$\Delta r_{i,\text{pred}} = 2,5|\dot{r}| \cdot T. \quad (6-13)$$

T ist die Zykluszeit, sodass Δr_{pred} die geschätzte zurückgelegte Entfernung zwischen zwei konsekutiven Zyklen angibt.

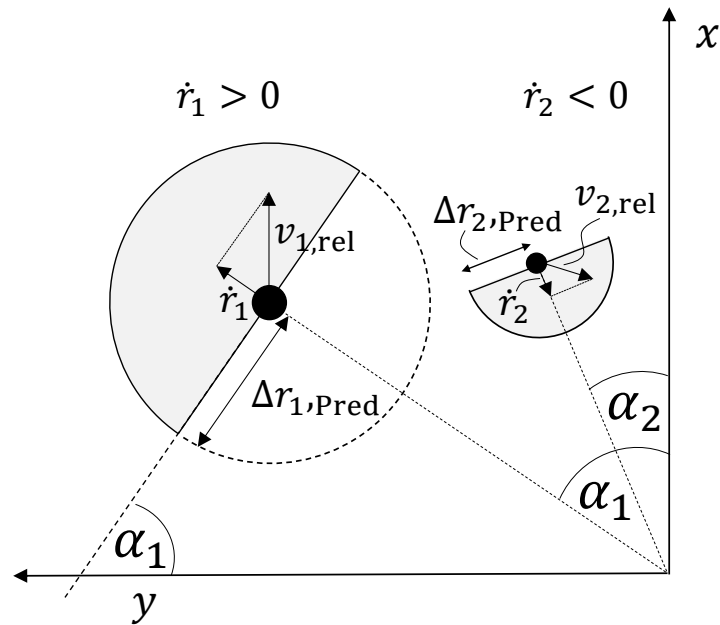


Abbildung 6-6: Resultierender Suchbereich der initialen Prädiktion

Dieser Suchbereich wird zur Datenassoziation genutzt. Liegt ein Cluster innerhalb dieses Suchbereichs, wird sie dem entsprechenden Objekt zugeordnet.

Die Filterung dieser Zuordnung nach einer initialen Prädiktion erfolgt ebenfalls abweichend vom EKF. Da keine spezifische Prädiktion erfolgte, kann diese nicht korrigiert werden, wie es beim EKF der Fall ist. Daher wird die Position des zugeordneten Clusters ohne Korrektur als Objektzustand angenommen. Besitzt dieses Cluster bereits die kartesischen Geschwindigkeiten, werden diese ebenfalls als Objektzustand übernommen. Sind die kartesischen Geschwindigkeiten allerdings nicht bekannt, erfolgt an dieser Stelle deren Bestimmung unter Betrachtung des initialen Objektzustands und des zugeordneten Clusters, wie in Kapitel 6.6.4 beschrieben wird. Die nun errechneten Geschwindigkeiten werden dem aktuellen Objektzustand hinzugefügt. Daraus lässt sich schließen, dass die initiale Prädiktion maximal einmal vorgenommen wird. Alle weiteren Prädiktionen dieses Objekts erfolgen mit dem EKF.

6.6.2 Prädiktion und Filterung mit dem Extended-Kalman-Filter

Das Extended Kalman Filter führt die Prädiktion und Filterung eines Objekts, dessen kartesischen Geschwindigkeiten bekannt sind, durch. Dieser nutzt zur Prädiktion die sogenannte Zustandsübergangsfunktion $f(\mathbf{S})$. Innerhalb dieser Arbeit wird das Constant Velocity Model als Bewegungsmodell genutzt. Untersuchungen im Verlauf der Arbeit zum Constant Turn Rate Model haben ergeben, dass die Unsicherheiten aus der Schätzung der kartesischen Geschwindigkeit und der Gierrate mit jedem Zyklus einer Objektverfolgung wachsen, sodass das Constant Velocity Model hier stabilere Ergebnisse liefert. Die Besonderheit des EKFs ist, dass nicht nur der Objektzustand \mathbf{S}_k prädiziert wird, sondern auch dessen Kovarianzmatrix \mathbf{P}_k . Der Zustandsvektor \mathbf{S}_k besteht aus den Positionen x und y sowie aus den Geschwindigkeiten v_x und v_y des Objekts im kartesischen Fahrzeugkoordinatensystem und lautet:

$$\mathbf{S}_k = \begin{bmatrix} x \\ y \\ v_{\text{rel},x} \\ v_{\text{rel},y} \end{bmatrix}, \quad (6-14)$$

Die Prädiktion der Zustandskovarianz basiert auf der Prozesskovarianz \mathbf{Q} , die das Rauschen des Bewegungsmodells beschreibt, und der Linearisierung der Zustandsübergangsfunktion. Die Linearisierung erfolgt durch die Jacobi-Matrix \mathbf{J}_f von $f(\mathbf{S}_k)$, sodass die Prädiktionsschritte für den Zyklus k lauten:

$$\mathbf{S}_k = f(\mathbf{S}_{k-1}), \quad (6-15)$$

$$\mathbf{P}_k = \mathbf{J}_f \mathbf{P}_{k-1} \mathbf{J}_f^T + \mathbf{Q}, \quad (6-16)$$

Die Zustandsübergangsfunktion $f(\mathbf{S})$ des Constant Velocity Model lautet dabei:

$$f(\mathbf{S}) = \begin{bmatrix} x + v_{\text{rel},x} \cdot T \\ y + v_{\text{rel},y} \cdot T \\ v_{\text{rel},x} \\ v_{\text{rel},y} \end{bmatrix} \quad (6-17)$$

Nach einer erfolgten Datenassoziation mit den prädizierten Größen, wie sie in Kapitel 6.4 beschrieben ist, folgt deren Korrektur. Dazu wird für jeden Zyklus der sogenannte Kalman Gain berechnet, welcher sich aus der Zustandskovarianz \mathbf{P}_k , der Messkovarianz \mathbf{R} und der linearisierten Messfunktion $h(\mathbf{S})$ ergibt. Die Linearisierung erfolgt hier ebenfalls durch die Jacobi-Matrix \mathbf{J}_h von $h(\mathbf{S}_k)$. Die Berechnung des Kalman Gains \mathbf{K}_k für den Zyklus k lautet wie folgt:

$$\mathbf{K}_k = \mathbf{P}_k \mathbf{J}_h^T (\mathbf{J}_h \mathbf{P}_k \mathbf{J}_h^T + \mathbf{R})^{-1} \quad (6-18)$$

Der prädizierte Objektzustand \mathbf{S}_k und die Zustandskovarianz \mathbf{P}_k werden korrigiert mit:

$$\mathbf{S}_k = \mathbf{S}_k + \mathbf{K}_k (\mathbf{y}_k - h(\mathbf{S}_k)) \quad (6-19)$$

$$\mathbf{P}_k = (\mathbf{E} - \mathbf{K}_k \mathbf{J}_h) \mathbf{P}_k \quad (6-20)$$

\mathbf{y}_k beschreibt dabei den Messvektor des zugeordneten Messwerts aus der erfolgten Datenassoziation.¹¹¹ Der Messvektor beinhaltet die Ausgangsgrößen des Radarsensors und lautet

$$\mathbf{y}_k = \begin{bmatrix} r \\ \alpha \\ \dot{r} \end{bmatrix}, \quad (6-21)$$

wobei r die radiale Entfernung, α den Azimutwinkel und \dot{r} die Dopplergeschwindigkeit beschreiben.

Der Term $\mathbf{y}_k - h(\mathbf{S}_k)$ in (6-19) berechnet die Differenz der Prädiktion zum Messwert. Dazu muss die Prädiktion mithilfe eines Vektors ausgedrückt werden, welcher analog zum Messwert ist. Dazu wird die Messfunktion $h(\mathbf{S})$ genutzt, welche lautet:

¹¹¹ Challa, S. et al.: Fundamentals of Object Tracking (2011), S. 36.

$$h(\mathbf{S}) = \begin{bmatrix} \sqrt{x^2 + y^2} \\ \arctan\left(\frac{y}{x}\right) \\ \sqrt{v_{\text{rel},x}^2 + v_{\text{rel},y}^2} \cdot \cos\left(\arctan\left(\frac{y}{x}\right) - \arctan 2(v_{\text{rel},y}, v_{\text{rel},x})\right) \end{bmatrix} = \begin{bmatrix} r \\ \alpha \\ \dot{r} \end{bmatrix} \quad (6-22)$$

Dabei entspricht $\arctan 2(v_{\text{rel},y}, v_{\text{rel},x})$ dem Gierwinkel des Objekts, da mit der Differenz des Gier- und Azimutwinkels die trigonometrische Beziehung zwischen der Relativ- und der Dopplergeschwindigkeit hergestellt wird. Diese Beziehung ist im folgenden Kapitel in Abbildung 6-8 verdeutlicht.

In der Gleichung (6-18) ist zu erkennen, dass \mathbf{K} proportional zur Zustandskovarianz \mathbf{P} und antiproportional zur Messkovarianz \mathbf{R} ist. Das bedeutet, dass der Kalman Gain mit steigender Unsicherheit des Zustandes \mathbf{P} wächst, sodass im Korrekturschritt die Differenz zwischen Prädiktion und zugeordnetem Messwert einen höheren Einfluss hat. Der zugeordnete Messwert wird damit höher gewichtet als die Prädiktion. Der umgekehrte Fall tritt ein bei einer hohen Messkovarianz \mathbf{R} , also einem hohen Sensorrauschen, sodass die Prädiktion höher gewichtet ist. Die korrigierte Größe \mathbf{S}_k aus (6-19) ist damit der Objektzustand für den aktuellen Zyklus k und wird im kommenden Zyklus erneut prädiziert und zur Datenassoziation genutzt.

Die Prozesskovarianzmatrix \mathbf{Q} beschreibt, wie bereits erwähnt, die Unsicherheiten im Bewegungsmodell, d.h. \mathbf{Q} gibt an, welche Sprünge in der Bewegung eines Modells zwischen zwei Zyklen auftreten können. Diese werden durch Beschleunigungen oder andere Störgrößen, wie z.B. Seitenwind, verursacht. Die Prozesskovarianz ergibt sich damit aus einer möglichen Störgröße σ_a^2 und der Zykluszeit T zu:¹¹³

$$\mathbf{Q} = \sigma_a^2 \cdot \begin{bmatrix} \frac{T^4}{4} & 0 & \frac{T^3}{2} & 0 \\ 0 & \frac{T^4}{4} & 0 & \frac{T^3}{2} \\ \frac{T^3}{2} & 0 & T^2 & 0 \\ 0 & \frac{T^3}{2} & 0 & T^2 \end{bmatrix} \quad (6-23)$$

Die Messkovarianz \mathbf{R} beschreibt das Sensorrauschen und wird in Abhängigkeit der Sensorgenauigkeiten σ_r , σ_α und $\sigma_{\dot{r}}$ gewählt.^{112,113}

$$\mathbf{R} = \begin{bmatrix} k_r \cdot \sigma_r^2 & 0 & 0 \\ 0 & k_\alpha \cdot \sigma_\alpha^2 & 0 \\ 0 & 0 & k_{\dot{r}} \cdot \sigma_{\dot{r}}^2 \end{bmatrix} \quad (6-24)$$

Die Faktoren k_r , k_α und $k_{\dot{r}}$ der Messkovarianz sowie die Größe σ_a der Prozesskovarianz wurden innerhalb dieser Arbeit mit der Auswertung von jeweils 417 unterschiedlichen

¹¹² Diese sind im Datenblatt des genutzten Sensors für jeden Messwert vorzufinden.

¹¹³ Balzer, P.: Das Kalman-Filter (2016).

Parameterkombinationen für fünf verschiedene Szenarien bestimmt. Die Szenarien umfassten Abläufe mit konstanten Geschwindigkeiten und beschleunigten Bewegungen sowie Gierbewegungen der Objekte. Die Auswertung erfolgte mit der GOSPA-Metrik für Tracking-Algorithmen. Diese Metrik bewertet ein Tracking-Ergebnis hinsichtlich der Genauigkeit der Positions- und Geschwindigkeitsschätzung und der geschätzten Zahl von Objekten zu jedem Zyklus.¹¹⁴ Die Wertebereiche der zu definierenden Größen wurden mithilfe von stichprobenartiger Auswertung auf zehn diskrete Werte beschränkt, welche in Tabelle 4 zu finden sind. Daraus wurde nicht jede Parameterkombination getestet, da die Zahl aller möglichen Variationen aus den Wertebereichen bei 10^4 pro Szenario liegt und damit den zeitlichen Rahmen dieser Arbeit übersteigt. Die Kombinationen wurden stattdessen so beschränkt, dass nur ein Parameter innerhalb seines Wertebereichs variiert wird, während die restlichen Größen einen festen Wert behalten. Dieser feste Wert ist einmal der Niedrigste und einmal der Höchste des jeweiligen Wertebereichs. Ausnahme ist die Größe σ_a , welche für jede Variation ebenfalls vollständig durchvariiert wurde, da die gesamte Prozesskovarianzmatrix auf dieser einen Größe basiert und damit eine Variation einen nicht vernachlässigbaren Einfluss auf das EKF hat. Die Ergebnisse der 417 Parametervariationen pro Szenario wurden so zusammengefasst, dass die Kombination mit dem insgesamt besten Ergebnis identifiziert wurde. Diese finalen Werte der Parameter sind ebenfalls in der nachfolgenden Tabelle zu finden. Eingesetzt in \mathbf{Q} und \mathbf{R} aus (6-23) und (6-24) ergeben sie die endgültige Form der Prozesskovarianz- und Messkovarianzmatrix des genutzten Extended Kalman Filters.

¹¹⁴ Rahmathullah, A. S. et al.: Generalized optimal sub-pattern assignment metric (2017).

Tabelle 4: Gewählte Wertebereiche und finale Werte der Parameter des EKF

Parameter	Wertebereich	Finaler Wert nach Auswertung
σ_a	$\{1, 2, 3, 4, 5\}$	2
k_r	$\{1, 2, 3, 4, 5, 6, 7, 9, 10\}$	1
k_α	$\{1, 2, 3, 4, 5, 6, 7, 9, 10\}$	1
$k_{\dot{r}}$	$\{1, 10, 50, 100, 200, 300, 500, 700, 800, 1000\} \cdot$	1000

6.6.3 Eigenbewegungskompensation

Da das Egofahrzeug in der Regel ebenfalls in Bewegung ist, berücksichtigt eine Eigenbewegungskompensation der prädizierten Objektzustände die Kinematik des Egofahrzeugs zwischen zwei Messzyklen. Die Eigenbewegungskompensation erfolgt durch eine einfache Koordinatentransformation der prädizierten Werte, sodass das Fahrzeugkoordinatensystem vom Zeitpunkt $k - 1$ in das aktuelle Fahrzeugkoordinatensystem zum Zeitpunkt k transformiert wird. Die zugehörige Transformation ergibt sich lediglich aus einer Rotation. Eine Translation ist nicht notwendig, da die Geschwindigkeiten der Objekte bereits als Relativgeschwindigkeiten angegeben werden. Die Transformationsmatrix lautet mit der Gierrate $\dot{\phi}_{ego}$ des Egofahrzeugs:

$$T_{\text{rot}}^{k|k-1} = \begin{bmatrix} \cos(\dot{\phi}_{ego} \cdot T) & \sin(\dot{\phi}_{ego} \cdot T) & 0 & 0 \\ -\sin(\dot{\phi}_{ego} \cdot T) & \cos(\dot{\phi}_{ego} \cdot T) & 0 & 0 \\ 0 & 0 & \cos(\dot{\phi}_{ego} \cdot T) & \sin(\dot{\phi}_{ego} \cdot T) \\ 0 & 0 & -\sin(\dot{\phi}_{ego} \cdot T) & \cos(\dot{\phi}_{ego} \cdot T) \end{bmatrix} \quad (6-25)$$

$\dot{\phi}_{ego} \cdot T$ beschreibt mit der Zykluszeit T die Rotation des Egofahrzeuges zwischen den konsekutiven Zyklen $k - 1$ und k . Die eigenbewegungskompensierten Prädiktionen ergeben sich damit aus:

$$\mathbf{s}_k^k = \mathbf{T}_{\text{rot}}^{k|k-1} \mathbf{s}_k^{k-1} \quad (6-26)$$

\mathbf{s}_k^k und \mathbf{s}_k^{k-1} beschreiben dabei beide die gleiche Prädiktion, jedoch ausgedrückt in unterschiedlichen Koordinatensystemen.

6.6.4 Schätzung der kartesischen Geschwindigkeiten

Die Schätzung der kartesischen Geschwindigkeiten und damit des Gierwinkels ist Bestandteil der Filterung der Objektzustände. Dazu sind im Tracking-Algorithmus zwei unterschiedliche Methoden implementiert, welche nachfolgend erläutert werden. Im Anschluss daran werden die Kriterien zur Methodenwahl dargestellt.

6.6.4.1 Methode 1: Instantaneous Velocity Estimation

Die erste Methode des Instantaneous Velocity Estimation (IVE) basiert auf der Annahme, dass alle Detektionen desselben Objekts mit ihrer jeweiligen Dopplergeschwindigkeit den radialen Anteil der gleichen relativen Geschwindigkeit angeben.¹¹⁵ Dieser radiale Anteil variiert je nach Azimutwinkel der Detektion. Damit lässt sich die Dopplergeschwindigkeit \dot{r} als Ausdruck der kartesischen Komponenten $v_{\text{rel},x}$, $v_{\text{rel},y}$ und des Azimutwinkel α formulieren:

$$\dot{r} = v_{\text{rel},x} \cdot \cos(\alpha) + v_{\text{rel},y} \cdot \sin(\alpha) \quad (6-27)$$

Diese Beziehung ist in Abbildung 6-7 dargestellt. Darin ist die relative Geschwindigkeit v_{rel} einer Detektion mit dem Azimutwinkel α aufgeteilt in seine Komponenten $v_{\text{rel},x}$ und $v_{\text{rel},y}$. Die Anteile $v_{\text{rel},x} \cdot \cos(\alpha)$ und $v_{\text{rel},y} \cdot \sin(\alpha)$ sind in \dot{r} farbig hinterlegt und zeigen damit den Zusammenhang dieser Größen. Da $v_{\text{rel},x}$ und $v_{\text{rel},y}$ in der Gleichung (6-27) zwei Unbekannte sind, lässt sich diese nicht einfach lösen, sodass ein lineares Gleichungssystem mit mindestens zwei Gleichungen aufgestellt wird, welches die kartesischen Geschwindigkeiten liefert. Je Detektion ergibt sich eine Gleichung, sodass für n Detektionen gilt:¹¹⁶

$$\begin{aligned} \dot{r}_1 &= v_{\text{rel},x} \cdot \cos(\alpha_1) + v_{\text{rel},y} \cdot \sin(\alpha_1) \\ \dot{r}_2 &= v_{\text{rel},x} \cdot \cos(\alpha_2) + v_{\text{rel},y} \cdot \sin(\alpha_2) \\ &\dots \\ \dot{r}_n &= v_{\text{rel},x} \cdot \cos(\alpha_n) + v_{\text{rel},y} \cdot \sin(\alpha_n) \end{aligned} \quad (6-28)$$

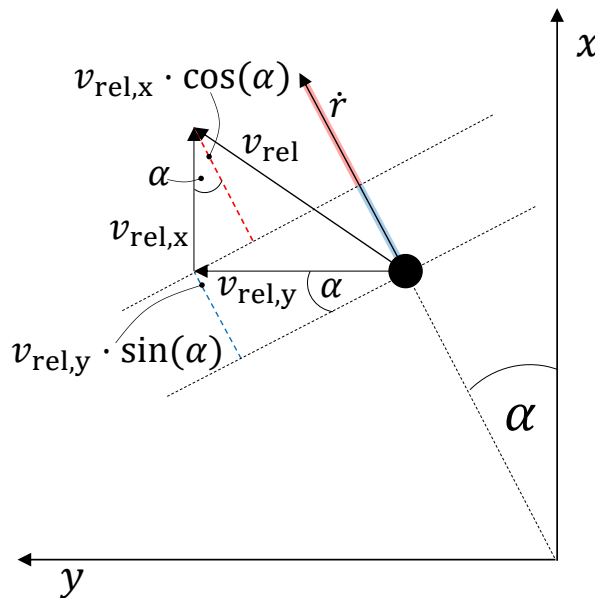


Abbildung 6-7: Darstellung der Beziehung zwischen Dopplergeschwindigkeit und den kartesischen Komponenten der relativen Geschwindigkeit

¹¹⁵ Da kein hochauflösender Radarsensor genutzt wird, können minimale Abweichungen zwischen den relativen Geschwindigkeiten der Detektionen vernachlässigt werden.

¹¹⁶ Gosala, N. B.; Meng, X.: An RLS-Based Instantaneous Velocity Estimator for Extended Radar Tracking (2020), S. 3–4.

Das in (6-28) beschriebene Gleichungssystem findet nicht nur in der Filterung Verwendung, sondern auch in der Bestimmung der kartesischen Geschwindigkeiten von Clustern vor der Datenassoziation. Sind im Cluster mehr als eine Detektion vorhanden, wird geprüft, ob diese Methode für dieses Cluster anwendbar ist. Zum einen wird der Rang des aufgestellten Gleichungssystem geprüft, welcher mindestens 2 betragen muss. Zum anderen wird der Einfluss der Abweichungen in der relativen Geschwindigkeit der Detektionen betrachtet. Sind die Abweichungen zu hoch, versagt die Methode und liefert fehlerhafte Ergebnisse. Die Identifikation des Einflusses erfolgt durch eine paarweise Prüfung der Detektionen. Das Gleichungssystem wird auf alle Paarkombinationen angewendet, sodass anschließend die Ergebnisse der kartesischen Geschwindigkeiten untereinander verglichen werden. Finden sich hier Ausreißer, versagt die Methode und wird nicht auf das Cluster angewendet. Die kartesischen Geschwindigkeiten bleiben unbestimmt.

In der Filterung der Objektzustände findet die Bestimmung der kartesischen Geschwindigkeiten des zugeordneten Clusters statt, sofern diese nicht, wie oben beschrieben, bereits bestimmt werden konnten. Nach einer erfolgten Datenassoziation stehen mehr Informationen zur Verfügung, sodass die eine Detektion des zugeordneten Clusters durch den vorhandenen Objektzustand ergänzt wird. Die Informationen des vorherigen Zyklus werden damit mit den Detektionen des aktuellen Zyklus verknüpft. Um den Objektzustand in (6-28) zu integrieren, wird dieser zunächst mithilfe der Messfunktion $h(S)$ als Messwert abgebildet, sodass der Azimutwinkel und die Dopplergeschwindigkeit bekannt werden. Aus diesen Größen und denen des Clusters wird das Gleichungssystem aus (6-28) aufgesetzt und nach v_x und v_y aufgelöst, sofern der Rang des Gleichungssystem gleich 2 ist.

Vorteil dieser Methode ist die Unabhängigkeit der Detektionen von einem Referenzpunkt am Objekt. Allerdings treten bei kleineren Abweichungen hohe Fehler in der Berechnung auf, sodass als Alternative die folgende Methode genutzt wird.

6.6.4.2 Methode 2: Gierwinkelschätzung aus Positionsänderung

Die zweite Methode zur Bestimmung der kartesischen Geschwindigkeiten basiert auf der Schätzung des Gierwinkels anhand der Positionsänderung zwischen dem zugeordneten Cluster des aktuellen Zyklus und dem Objektzustand des vorherigen Zyklus. Dazu werden die Mittelpunkte beider Bounding-Boxen als Referenz genutzt. Wie die Bounding-Boxen geschätzt werden, ist in Kapitel 6.6.5 erläutert. Der Gierwinkel ist dabei der Winkel zwischen der x-Achse des Fahrzeugkoordinatensystem und dem Vektor der euklidischen Distanz, welcher die Positionsänderung angibt. Im Gegensatz zur Instantaneous Velocity Estimation ist diese Methode nicht unabhängig von einem Referenzpunkt am Objekt. Da dieser aber nicht für alle Detektionen übereinstimmt, entstehen so Ungenauigkeiten in den Ergebnissen dieser Methode.

Ein Beispiel ist in Abbildung 6-8 dargestellt. Zu sehen sind nummerierte Markierungen (1 und 2) für die Objektposition $(x_{t_0}|y_{t_0})$ zum Zeitpunkt t_0 und für die Bounding-Box-Position $(x_{t_1}|y_{t_1})$ des Clusters zum Zeitpunkt t_1 . Der Gierwinkel ist mit jeweils φ_i eingezeichnet.

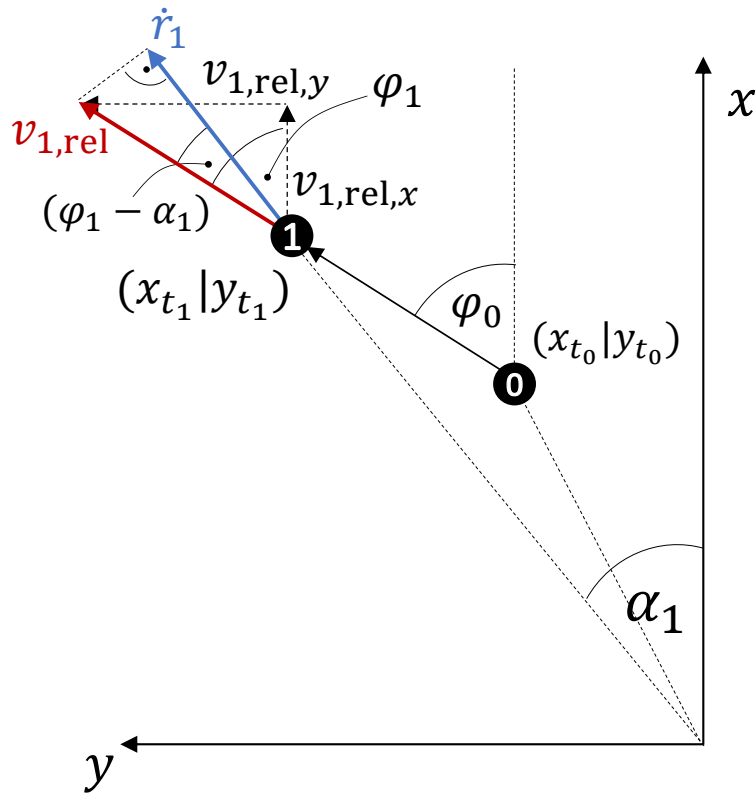


Abbildung 6-8: Gierwinkel φ aus Positionsänderung

Der Gierwinkel φ_1 ergibt sich dann aus:

$$\varphi_1 = \arctan2((y_{t_1} - y_{t_0}), (x_{t_1} - x_{t_0})) \quad (6-29)$$

Daraus lässt sich mithilfe der Dopplergeschwindigkeit \dot{r}_{t_1} und des Azimutwinkels α_1 die kartesischen Geschwindigkeiten $v_{1,x}$ und $v_{1,y}$ ermitteln:

$$v_{1,rel} = \frac{\dot{r}_{t_1}}{\cos(\varphi_1 - \alpha_1)} \quad (6-30)$$

$$v_{1,rel,x} = v_{1,rel} \cdot \cos(\varphi_1) \quad (6-31)$$

$$v_{1,rel,y} = v_{1,rel} \cdot \sin(\varphi_1) \quad (6-32)$$

Setzt man (6-30) in jeweils (6-31) und (6-32) ein, folgen die finalen Formeln zu den kartesischen Geschwindigkeiten:

$$v_{1,rel,x} = \dot{r}_{t_1} \cdot \frac{\cos(\varphi_1)}{\cos(\varphi_1 - \alpha_1)} \quad (6-33)$$

$$v_{1,rel,y} = \dot{r}_{t_1} \cdot \frac{\sin(\varphi_1)}{\cos(\varphi_1 - \alpha_1)} \quad (6-34)$$

Diese Geschwindigkeiten werden dem Cluster zugordnet und als Objektzustand übernommen.

6.6.4.3 Kriterien zur Methodenwahl

Die Wahl der Methode zur Berechnung der kartesischen Geschwindigkeiten basiert auf zwei Punkten, welche im Ablaufdiagramm in Abbildung 6-9 dargestellt sind. An erster Stelle wird geprüft, ob die Methode Instantaneous Velocity Estimation auf die Zusammensetzung der Detektionen anwendbar ist. Dazu wurde die Prüfweise in 6.6.4.1 erläutert. Ist diese nicht anwendbar, erfolgt die Anwendung der zweiten Methode der Gierwinkelschätzung anhand der Positionsänderung. Sind allerdings beide Methoden akzeptabel, werden beide Methoden auf die Detektionen angewandt und das Ergebnis mit dem kleineren Wert für v_y gewählt. Da durch das Verknüpfen zweier Zeitpunkte beide Methoden fehleranfällig sind, insbesondere in der Berechnung der y-Komponente der kartesischen Geschwindigkeiten, wird hier das Wertepaar genutzt, welches durch einen niedrigeren Wert für v_y plausibler scheint. Fehler in der Berechnung der Geschwindigkeiten sind hier allerdings dennoch nicht auszuschließen.

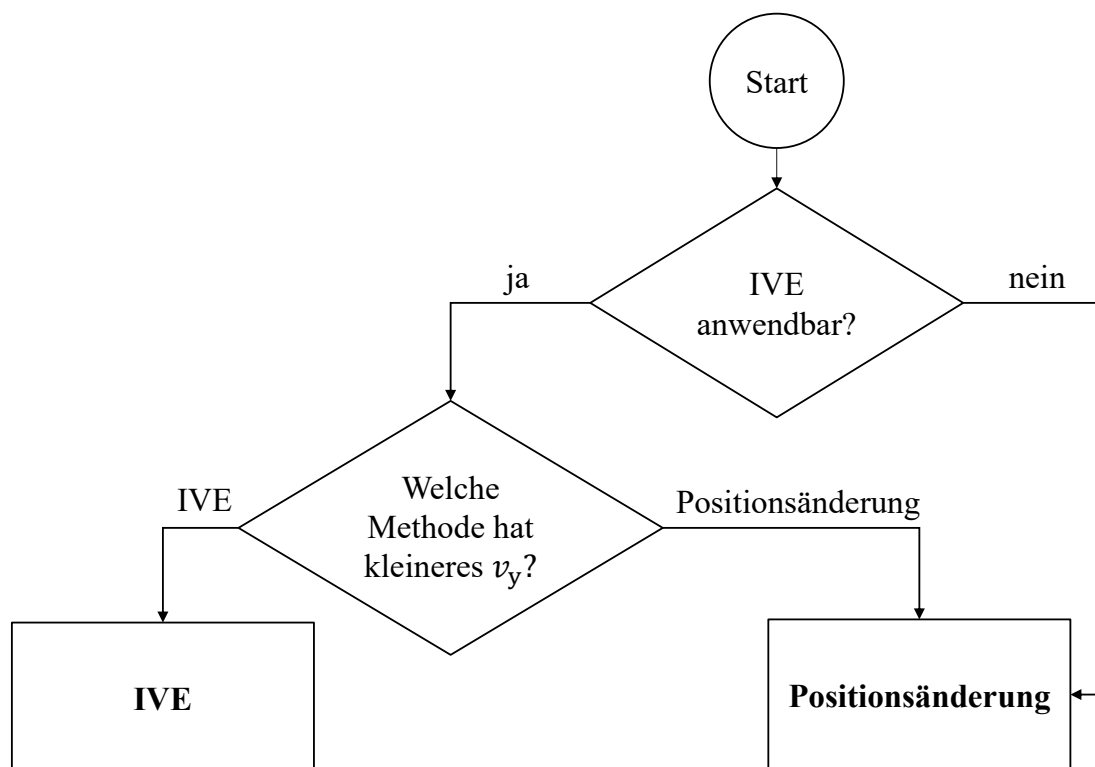


Abbildung 6-9: Ablaufdiagramm zur Wahl der Berechnungsmethode

6.6.5 Schätzung der Bounding-Box

Die Position eines Objekts oder Clusters ist der Mittelpunkt der Bounding-Box, welche um die entsprechenden Detektionen gelegt wird. Zum einen sollen Bounding-Box Informationen über die Dimensionen des detektierten Objekts liefern und zum anderen die Position des Objekts näher spezifizieren. Dabei wird je nach Konstellation der Detektionen unterschiedlich vorgegangen. Bei Konstellationen aus nur einer Detektion gibt es keinerlei Anhaltspunkte über die Objektdimensionen. Daher wird innerhalb dieser Arbeit eine Mindestgröße der Objekte angenommen, welche bei einer Breite

von 180 cm und einer Länge von 430 cm liegt.¹¹⁷ Objekte wie Fußgänger oder Zweiradfahrer werden damit überdimensioniert. Allerdings ist hinsichtlich einer allgemeinen Einschätzung der Fahrsituation eine Überdimensionierung der Verkehrsteilnehmer einer Unterdimensionierung vorzuziehen, um die Kollisionsgefahr zu verringern. Diese Mindestmaße werden an der einen vorhandenen Detektion so angelegt, dass diese in der Fahrzeuglängsachse sitzt und mit dem Fahrzeugheck abschließt, wie es in der Abbildung 6-10 am mittleren Objekt zu sehen ist.

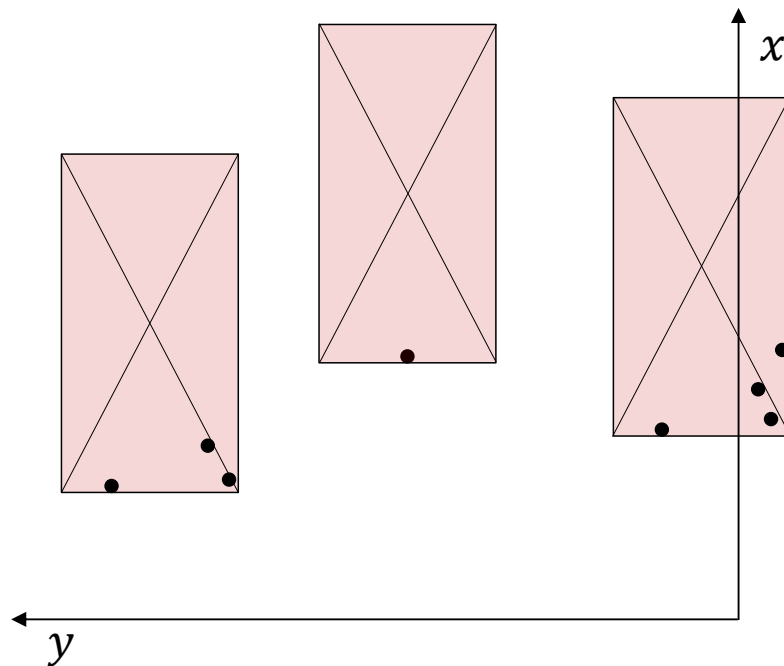


Abbildung 6-10: Beispielhafte Schätzung der Bounding-Box

Sind mehrere Detektionen vorhanden, wird geprüft, ob die Distanzen in x- und y-Richtung durch die Verteilung der Detektionen die genannten Mindestmaße erreicht. Ist dies in beide Richtungen der Fall, legt die Methode eine Bounding-Box so um die Detektionen, dass alle von dieser umschlossen werden. Erreicht die Verteilung der Detektionen nur die Mindestbreite, nicht aber die Mindestlänge, wird eine Bounding-Box mit der erfassten Objektbreite und der definierten Mindestlänge um die Detektionen gelegt. Die dem Egofahrzeug nächste Detektion schließt dabei mit dem Fahrzeugheck ab.

Erreicht die Verteilung der Detektionen nicht die definierte Mindestbreite, folgt die Prüfung des lateralen Wertebereichs der Detektionen. Liegen alle Detektionen im positiven y-Bereich, wird die Detektion mit dem kleinsten y-Wert als rechte Außenkante der Bounding-Box angenommen. Dieses Beispiel ist in der Abbildung auf der linken Seite zu sehen. Befinden sich alle Detektionen wiederum im negativen y-Bereich, wird die Detektion mit dem größten y-Wert, die also dem Egofahrzeug am nächsten ist, als linke Außenkante der Bounding Box angenommen. Befinden sich die Detektionen sowohl im positiven als auch im negativen y-Bereich, wird mithilfe des Mittelwerts aller y-Punkte die Verteilung der Detektionen ermittelt. Befindet sich nun die Detektion mit dem maximalen y-Wert näher am Mittelwert als die Detektion mit dem minimalen y-Wert, wird der maximale y-Wert

¹¹⁷ Diese Maße orientieren sich an einem Kompaktklassewagen.

als linke Außenkante der Bounding-Box angenommen. Liegt die Detektion mit dem minimalen y-Wert näher am Mittelwert, wird dieser als rechte Außenkante der Bounding-Box angenommen. Dieses Beispiel wird in der Abbildung am rechten Objekt gezeigt.

Die Mittelpunkte der geschätzten Bounding-Boxen für Cluster und Objekte, werden jeweils als Referenzpunkt zur Datenassoziation, zur Schätzung des Gierwinkels sowie zur Löschung von Duplikaten genutzt

6.7 Löschung von Duplikaten

Die Klassifikation mehrerer Objekte als Duplikate erfolgt, sobald ihre Positionen weniger als $\Delta d = 2 \text{ m}$ voneinander entfernt sind sowie ihre relativen kartesischen Geschwindigkeiten sich um weniger als $\Delta v_{\text{rel},x} = 2 \frac{\text{m}}{\text{s}}$ und $\Delta v_{\text{rel},y} = 2 \frac{\text{m}}{\text{s}}$ voneinander unterscheiden. Finden sich innerhalb eines Zyklus Duplikate, wird davon ausgegangen, dass diese das gleiche Objekt verfolgen, sodass die Objekte mit der kürzeren Lebenszeit gelöscht werden.

6.8 Zusammenfassung

Dieses Kapitel beschreibt umfassend die implementierten Methoden des Tracking-Algorithmus. Dazu wurde zu Beginn das Verfahren zur Filterung der Detektionen mithilfe der abstandskompensierten Signalleistung vorgestellt. So werden Detektionen, denen keine Relevanz zugeschrieben wird, für den weiteren Verlauf im Tracking-Algorithmus vernachlässigt. Anschließend folgte die Umrechnung der Bin-Daten der gefilterten Detektionen in polare und kartesische Koordinaten. Dazu wurden die einzelnen Umrechnungsverfahren für die Entfernung, den Azimutwinkel und insbesondere für die Dopplergeschwindigkeit aufgezeigt. Während die Umrechnung der Entfernung und des Azimutwinkel als simpel einzustufen sind, stellt die Berechnung der Dopplergeschwindigkeit aufgrund der Mehrdeutigkeiten der Fouriertransformation ein komplexeres Verfahren dar. Da der Tracking-Algorithmus hier auf die Werte der kommenden Zyklen zurückgreift, erzeugt das implementierte Berechnungsverfahren einen Verzug von zwei Zyklen.

Des Weiteren folgte in diesem Kapitel die Erläuterung der Clustering-Methode, welche in zwei Phasen geteilt ist. Dieses basiert nicht auf bereits errechneten Dopplergeschwindigkeiten, sondern berücksichtigt die Dopplerbins. Daher können an dieser Stelle nach erfolgter Zusammenfassung Lücken aus der Berechnung der Dopplergeschwindigkeit gefüllt werden. Die Clustering-Methode verwirft damit Detektionen, denen auch hier keine Geschwindigkeit zugeordnet werden konnte, für den weiteren Verlauf des Tracking-Algorithmus. Auf das Verfahren der Datenassoziation, welches auf der Methode des Global Nearest Neighbor basiert, wurde anschließend eingegangen. Erfolgte in der Datenassoziation die Zuordnung eines Clusters zu einem Objekt, passt das Track-Management den Status des Objektes an. Die dazu implementierte Methode wird anhand eines Schaubildes, welches die Zustandsübergänge graphisch darstellt, erläutert. Die umfassende Beschreibung der nächsten Schritte, die Prädiktion und Filterung, folgte anschließend. Dabei wird in diesen Schritten zwischen zwei Prädiktions- und Filterungsarten unterschieden. In Abhängigkeit der kartesischen Geschwindigkeiten erfolgt entweder eine initiale Prädiktion und Filterung oder eine Prädiktion und Filterung auf

Basis des Extended Kalman Filter, welches das Constant Velocity Model nutzt. Die initiale Prädiktion resultiert in einem Prädiktionsbereich, während die Prädiktion des EKF eine diskrete Angabe macht. Zusätzlich wurden in diesem Abschnitt die Methoden zur Bestimmung der kartesischen Geschwindigkeiten nach der ersten Datenassoziation erläutert. Dabei folgen je nach Detektionskonstellation die Anwendung zweier unterschiedlicher Verfahren. Da die Datenassoziation und Filterung Objekt- und Clusterpositionen anhand der zugehörigen Bounding-Box verwenden, schließt das Kapitel mit der Schätzung der Bounding-Box und anschließend mit der Löschung von Objektduplikaten ab.

7 Evaluierung

In diesem Kapitel erfolgt die Evaluierung des implementierten Tracking-Algorithmus. Dazu werden zuerst die Grundlagen zur Leistungsbewertung von Trackern dargestellt und auf die verwendeten Kriterien (7.1) eingegangen. Im Anschluss folgen die Ergebnisse der Identifikation der Leistungsfähigkeit mit Referenz zur Industrielösung (7.2). Das Kapitel schließt mit dem Vergleich der Tracking-Ergebnisse zwischen Simulations- und Messdaten (7.3) ab, um die Übertragbarkeit der Ergebnisse zu untersuchen.

7.1 Leistungsbewertung von Tracking-Algorithmen

Die Leistungsbewertung von Tracking-Algorithmen erfolgt anhand mehrerer Kriterien. Diese lassen sich unterteilen in solche, die zum einen den zeitlichen Verlauf des Tracking-Algorithmus betrachten und zum anderen in jene, die Genauigkeit des Tracking-Ergebnis bewerten. Zusätzlich betrachtet die Leistungsbewertung die Kontinuität der Object-Tracks.¹¹⁸ Im weiteren Verlauf werden die einzelnen Kriterien näher erläutert.

Zeitliche Kriterien

Die zeitlichen Kriterien zur Bewertung eines Tracking-Algorithmus beziehen sich auf die Zeitspanne, die nötig ist, um entweder ein neues Objekt zu initiieren oder ein verschwundenes Objekt zu löschen. Die benötigte Zeitspanne für beide Fälle unterliegt jeweils einem Zielkonflikt:

- Ein neues Objekt soll so schnell wie möglich erkannt werden. Allerdings soll gleichzeitig die Initiierung von Geisterobjekten, den „False-Positives“, vermieden werden.
- Ein verschwundenes Objekt soll so schnell wie möglich gelöscht werden. Allerdings soll der Einfluss von vereinzelt fehlenden Messungen, den „False-Negatives“, durch Messfehler oder Verdeckung minimiert werden.

Diese Zielkonflikte werden in der Implementierung des Track-Management berücksichtigt. Hier wird der Status eines Objekts im Verlauf mehrerer konsekutiver Zyklen betrachtet, bevor es entweder bestätigt oder gelöscht wird.

Genauigkeit der Ergebnisse

Die Genauigkeit der Ergebnisse bewertet die Positions- und Geschwindigkeitsschätzung der verfolgten Objekte. Dabei werden die geschätzten Objektzustände mit den realen Zuständen der verfolgten Objekte verglichen. Ungenauigkeiten ergeben sich unter anderem durch Messfehler des Sensors und Unsicherheiten in der Schätzung der kartesischen Geschwindigkeiten. Zusätzlich werden Ungenauigkeiten durch das Bewegungsmodell und die dort getroffenen Annahmen, wie beispielsweise eine konstante Geschwindigkeit, sowie durch die festgelegten Kovarianzen des Extended Kalman Filters verursacht. Um die Genauigkeit beider Größen zu bewerten, wird der Fehler als euklidische Distanz

¹¹⁸ Ristic, B. et al.: A Metric for Performance Evaluation of Multi-Target Tracking Algorithms (2011), S. 3453.

zwischen der geschätzten und realen Position bzw. zwischen den geschätzten und realen kartesischen Geschwindigkeiten gemessen. Die Genauigkeit der Ergebnisse hat Einfluss auf die Trennfähigkeit zweier Objekte durch den Tracking-Algorithmus. Um die Trennfähigkeit zu bewerten, werden die benötigte laterale Distanz und die Geschwindigkeitsdifferenz zwischen zwei Zielobjekten zur separaten Verfolgung identifiziert.

Kontinuität der Objekttracks

Neben dem zeitlichen Verlauf bezüglich der Initiierung und Löschung von Objekten und der Genauigkeit der Ergebnisse wird die Stetigkeit der Objekttrajektorien betrachtet. Dabei wird eine kontinuierliche Objektverfolgung angestrebt, sodass die geschätzten Zustände zu jedem Zyklus bekannt sind. Zusätzlich soll eine Umbenennung der Tracks, also ein Wechsel der Track-ID, vermieden werden. Lücken in der Verfolgung entstehen durch fehlerhafte Löschung von Objekten oder Fehler in der Datenassoziation, z.B. das Vertauschen von Objekten.

7.2 Vergleich der Leistungsfähigkeit mit der Industrielösung

Der Vergleich der Leistungsfähigkeit des implementierten Tracking-Algorithmus mit der am Fachgebiet vorliegenden Industrielösung erfolgt auf Basis mehrerer Referenzszenarien. Diese Referenzszenarien prüfen unterschiedliche Kriterien der Leistungsbewertung. Da die Industrielösung nur reale Messdaten verarbeitet, wird in diesem Abschnitt der Leistungsvergleich nur mit dieser Art von Daten durchgeführt.

Um die geschätzten Zustände beider Tracker mit realen Daten vergleichen zu können, stehen die tatsächlichen Kinematik- und Positionsdaten des Zielfahrzeuges zur Verfügung. Dieses wurde dazu mit einer Hardware ausgestattet ist, welche in der Lage ist, diese Größen mithilfe von GPS-Daten und Inertialsensoren zu ermitteln.

7.2.1 Initiierung und Löschung von Objekten

Zur Initiierung und Löschung von Objekten werden die Zeitspannen von der ersten Detektion des Objekts durch den Sensor bis zur Initialisierung und Bestätigung der Objekte durch den Tracker betrachtet. Da für die Industrielösung nur die finale Objektliste vorhanden ist, ist hier keine Unterscheidung zwischen initiierten und bestätigten Objekten möglich. Daher wird lediglich die Zeitspanne zwischen Detektion und vorhandenem Objekt, ob initiiert oder bestätigt ist unbekannt, identifiziert. In den Referenzszenarien für die zeitlichen Kriterien wird ein Objekt verfolgt, dass zum einen in das Sensorfeld eintritt und zum anderen das Sensorfeld verlässt. Die Initiierung und Löschung der Objekte erfolgt damit am Rande des Sensorfeldes. In der nachfolgenden Tabelle sind die Szenarienvläufe aus Sicht des Egofahrzeuges zusammengefasst.

Tabelle 5: Szenarienvläufe zur Bewertung der Track-Initiierung

Nr.	Verlauf
1	Das Zielfahrzeug tritt von der rechten Seite in das Sensorsichtfeld, fährt dabei in positive y-Richtung und macht anschließend eine Linkskurve.
2, 3, 4	Das Zielfahrzeug tritt von der linken Seite in das Sensorsichtfeld, fährt dabei in negative y-Richtung und macht anschließend eine Rechtskurve.
5, 6, 7, 8, 9	Das Zielfahrzeug tritt von der linken Seite in das Sensorsichtfeld, fährt dabei in negative y-Richtung und macht anschließend eine Linkskurve.
10	Das Zielfahrzeug überholt das Egofahrzeug von links, fährt dabei in positive x-Richtung und tritt auf der linken Seite in das Sensorsichtfeld.

In Abbildung 7-1 sind die unterschiedlichen Zeitspannen von der ersten Detektion bis zur Objekterkennung beider Tracker gegenübergestellt. Dazu ist an der y-Achse die jeweilige Messung gelistet, für die jeweils die drei Größen ermittelt werden. Der blaue Balken beschreibt die Zeitspanne der Objekterkennung durch die Industrielösung, der orangene Balken die Zeitspanne bis zur Initiierung und der graue Balken die Zeitspanne bis zur Bestätigung des Objekts durch den implementierten Tracking-Algorithmus. Die Zeitspanne bis zur Bestätigung des Objekts wird ab der ersten Detektion gemessen, sodass die Initiierungsphase hier ebenfalls vorhanden ist. Ist für eine Messung kein orangener Balken sichtbar, liegt das daran, dass ein Objekt mit der ersten Detektion initiiert wird.

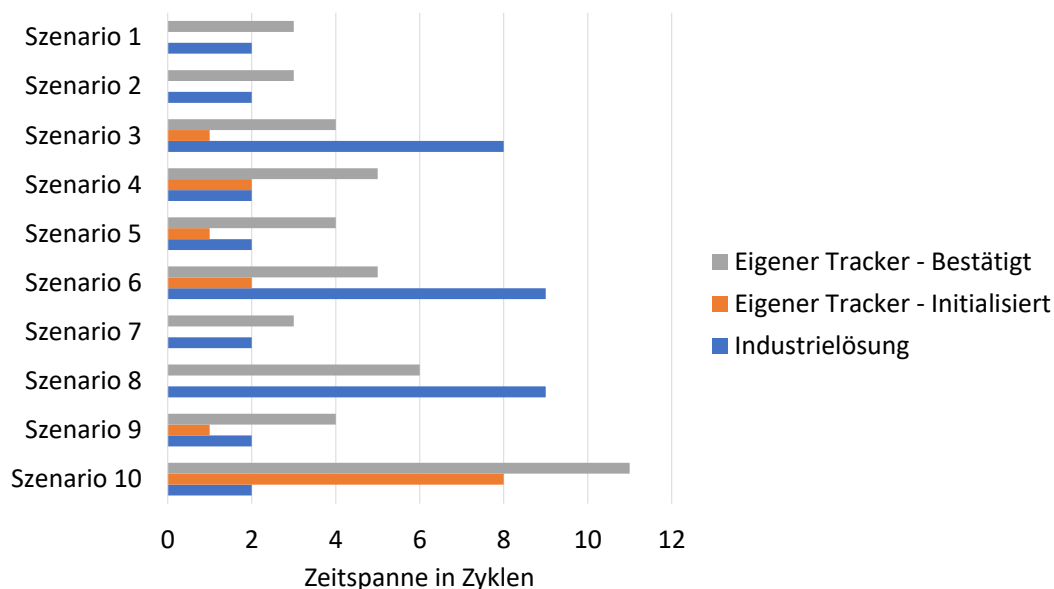


Abbildung 7-1: Vergleich der Zeitspannen bis zur Objekterkennung beider Tracker

Die Zeitspannen beider Tracker fallen je nach Messung unterschiedlich aus. Für die Industrielösung (blau) ist zu sagen, dass in 7 von 10 Messungen das Objekt zwei Zyklen nach der ersten Detektion verfolgt wird. Ist dies nicht der Fall, beginnt die Objektverfolgung nach 8 bzw. 9 Zyklen. Die Zeitspanne des eigenen Trackers hingegen schwankt stärker. Während die Zeitspanne bis zur Initiierung (orange) bei 9 von 10 Messungen zwischen null und zwei Zyklen liegt, variiert die Zeitspanne

bis zur Bestätigung (grau) zwischen 3 und 6 Zyklen. In 8 Fällen erfolgte die Bestätigung drei Zyklen nach der Initiierung. Szenario 10 sticht durch seine langen Zeitspannen von 8 und 11 Zyklen für den eigenen Tracker hervor. Die Unterschiede in den Zeitspannen z.B. für die Industrielösung in Szenario 8 und 9 mit jeweils 9 und 2 Zyklen sind nicht auf unterschiedliches Objektverhalten zurückzuführen, da diese Messdaten zwei Aufnahmen der gleichen Objekttrajektorien sind. Damit ist an dieser Stelle nicht genau geklärt, was die Ursachen für diese Abweichungen sind. Grundlegend ist aber der Trend zu erkennen, dass die Initiierung durch den eigenen Tracker genauso schnell oder schneller als die Objekterkennung der Industrielösung erfolgt. Die Bestätigung des Objekts durch den eigenen Tracker erfolgt allerdings erst nach einigen vergangenen Zyklen.

Analog dazu wurden die Zeitspannen für die Löschung der Objekte beim Austritt aus dem Sensorsichtfeld identifiziert. Hierzu eigneten sich allerdings nur vier Messungen, da der Tracker der Industrielösung nicht nur auf den Detektionen des Fernbereichssensor, wie er in Kapitel 2.2 dargestellt ist, sondern auch auf den Detektionen des Nahbereichssensors basiert. Somit hat das Verlassen des Sensorbereichs, auf dessen Basis diese Tracking-Algorithmus entwickelt wurde, keinen Einfluss auf den Tracker der Industrielösung. In der nachfolgenden Tabelle sind die Verläufe der verwendeten Szenarien zusammengefasst.

Tabelle 6: Szenarienvläufe zur Bewertung der Track-Löschung

Nr.	Verlauf
1	Das Zielfahrzeug kommt dem Egofahrzeug entgegen, fährt dabei in negative x-Richtung und verlässt das Sensorsichtfeld auf der linken Seite.
2, 3	Das Zielfahrzeug fährt in negative y-Richtung und verlässt das Sensorsichtfeld auf der rechten Seite.
4	Das Zielfahrzeug fährt in positive y-Richtung und verlässt das Sensorsichtfeld auf der linken Seite.

In Abbildung 7-2 sind die Ergebnisse der Szenarien dargestellt. Die Balken geben die Zeitspanne zwischen der letzten Detektion bis zur Löschung des Objekts an. Zu sehen ist, dass die Löschung der Objekte durch die Industrielösung nach einer unterschiedlichen Zahl an Zyklen erfolgt, welche zwischen 5 und 9 liegt. Die Löschung durch den eigenen Tracker variiert nur leicht und liegt für diese Zeitspanne zwischen 8 und 9 Zyklen. Da die Ergebnisse auf einer so kleinen Zahl von Messdaten basiert, lässt sich hier keine allgemeingültige Aussage treffen, allerdings deuten sie darauf hin, dass die Löschung der Objekte durch die Industrielösung früher erfolgt als durch den eigenen Tracker.

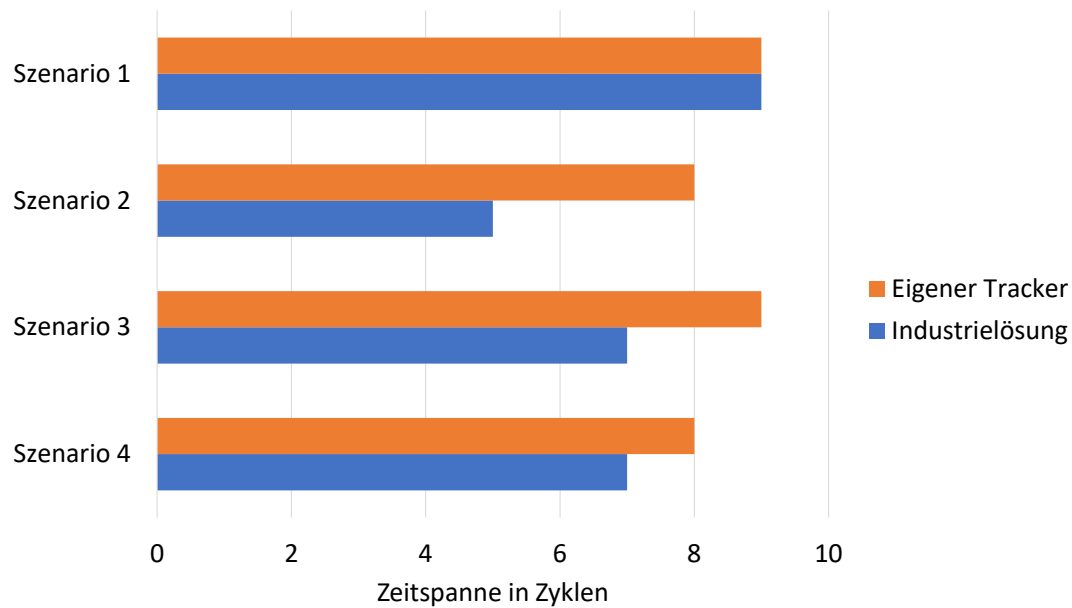


Abbildung 7-2: Vergleich der Zeitspannen bis zur Objektlöschung beider Tracker

Setzt man die identifizierten Zeitspannen beider Zustandsänderungen in Relation zu möglichen Relativgeschwindigkeiten, lässt sich für die Objektbestätigung in den vorhandenen Messdaten sagen, dass im Extremfall 11 Zyklen seit der ersten Detektion (Szenario 10) und damit 0,79 Sekunden vergangen sind.¹¹⁹ Hätte das verfolgte Objekt nun eine Relativgeschwindigkeit von $v_{\text{rel},x} = -111 \frac{\text{m}}{\text{s}}$, welches die maximal detektierbare Geschwindigkeit ist, befände es sich nach 0,79 Sekunden bereits $\Delta x = 88 \text{ m}$ näher am Egofahrzeug. Da die gesamte Länge des Sensorsichtfelds hier in weniger als 2 Sekunden durchfahren wäre und jeder vergangene Zyklus eine Annäherung von $\Delta x = 8 \text{ m}$ bedeutet, ist in diesem Fall eine frühestmögliche Bestätigung des Objekts anzustreben. Rechnet man nicht mit dem Extremfall, sondern mit den restlichen ermittelten Werten von 3 bis 6 Zyklen bis zur Bestätigung, ergeben sich bei einer Geschwindigkeit von $v_{\text{rel},x} = -111 \frac{\text{m}}{\text{s}}$ Annäherungen von $\Delta x = 24 \text{ m}$ bei 3 Zyklen und $\Delta x = 48 \text{ m}$ bei 6 Zyklen. Für $v_{\text{rel},x} = -55 \frac{\text{m}}{\text{s}}$ liegen diese Werte bei jeweils $\Delta x = 12 \text{ m}$ und $\Delta x = 24 \text{ m}$. Diese Werte stellen eine Situation dar, die weniger kritisch ist als der gezeigte Extremfall. Die benötigten Zyklen bis zur Bestätigung des Objekts lassen sich durch Parametervariationen im Track-Management beeinflussen. Daher ist bei der Entwicklung des Tracking-Algorithmus vorab das Anwendungsgebiet des Trackers zu definieren, oder das Track-Management adaptiv zu gestalten. So werden bei niedrigen Geschwindigkeiten, wie z.B. im Stadtverkehr, Objekte zu späteren Zyklen bestätigt als bei hohen Geschwindigkeiten. Diesen kritischen Situationen steht die Objektlöschung gegenüber. Eine verzögerte Löschung, wie es bei der Industrielösung und dem eigenen Tracker der Fall ist, ist einer fehlerhaften früheren Löschung vorzuziehen. So wird die Kollisionsgefahr, welche von einer möglichen Trajektoriengenerierung für das Egofahrzeug basierend auf den falschen Tracking-Ergebnissen ausgehen würde, reduziert.

¹¹⁹ Bei einer Zykluszeit von 0,072 Sekunden.

7.2.2 Geschwindigkeitsschätzung

Die Geschwindigkeitsschätzung ist Teil des Extended Kalman Filters. Die Referenzszenarien werden daher zwecks einer umfassenden Bewertung des Filters, so gewählt, dass aus funktionaler Sicht nicht nur der Bereich innerhalb der definierten Annahmen, sondern auch die Grenzbereiche und Bereiche darüber hinaus abgedeckt sind. Da die Prädiktion des EKF auf dem Constant Velocity Model basiert, werden an dieser Stelle Messdaten ausgewertet, die zum einen eine annähernd konstante Geschwindigkeit und zum anderen eine beschleunigte Bewegung beinhalten. In Tabelle 7 ist die Kinematik der verwendeten Szenarien zusammengefasst

Tabelle 7: Szenarien zur Auswertung der Geschwindigkeitsschätzung

Nr.	Szenario	Relative Geschwindigkeit	Relative Beschleunigung
1	Konstante Geschwindigkeit	-0,5 – 1 m/s	-2 – 2 m/s ²
2	Beschleunigtes Ziel	-7 – 7 m/s	-3 – 4 m/s ²
3	Beschleunigtes Ziel	0 – 33 m/s	0 – 5 m/s ²

Das erste verwendete Szenario weist eine annähernd konstante Geschwindigkeit auf, während das zweite Szenario einen größeren Geschwindigkeits- und auch Beschleunigungsbereich abdeckt. Das dritte Szenario mit einer hohen Beschleunigung von bis zu 5 m/s² wird genutzt, um das Verhalten des Filters, welches eine konstante Geschwindigkeit annimmt, zu untersuchen. Ziel ist hier, mit einer ausreichenden Korrektur der prädizierten Werte durch das Filter das Zielfahrzeug kontinuierlich zu verfolgen. Das Zielfahrzeug fährt in jedem Szenario stets mit einem Azimutwinkel von $\alpha = 0$ in Richtung der x-Achse des Fahrzeugkoordinatensystems. Daher wird zusätzlich zur Geschwindigkeitsschätzung im folgenden Kapitel auch der longitudinale und laterale Abstand untersucht. Ziel ist ebenfalls, diesen Abstand auch bei hohen Relativgeschwindigkeiten ausreichend gut einzuschätzen, um eine potentielle Trajektorienplanung anhand der Tracking-Ergebnisse zu ermöglichen.

Die beschriebenen Szenarien dieses und der nachfolgenden Kapitel wurden insgesamt dreimal durchgeführt und als Messdaten aufgenommen. Die im Anschluss diskutierten Ergebnisse sind qualitativ in allen Durchläufen zu erkennen. Zur Darstellung in den Diagrammen wurde jeweils das Szenario gewählt, in dem die beschriebenen Ergebnisse der Evaluierung am deutlichsten zu sehen sind.

In Abbildung 7-3 (oben) sind die Ergebnisse der Geschwindigkeitsschätzung des eigenen, implementierten Trackers sowie der dem FZD vorliegenden Industrielösung dargestellt. Zu sehen ist die geschätzte Relativgeschwindigkeit $v_{\text{rel},x}$ im Verlauf der Messzyklen t_{mess} . Zusätzlich ist die longitudinale Relativgeschwindigkeit des realen Objekts aufgetragen. Während der geschätzte Geschwindigkeitsbereich der Industrielösung sich wie die reale Geschwindigkeit ebenfalls auf -0,4 m/s und 1 m/s beschränkt, ist deutlich zu erkennen, dass die Geschwindigkeiten des implementierten Trackers stark zwischen -0,75 m/s und 1,5 m/s schwanken. Während nur der mittlere Verlauf der Schätzung sich dem Verlauf der realen Geschwindigkeit anpasst, ist die Schätzung der Industrielösung insgesamt deutlich stabiler. Dies lässt sich auch in der Darstellung des Fehlers in Abbildung 7-3 (unten) erkennen. Der generierte Fehler durch die Industrielösung liegt bei unter $|\Delta v_{\text{rel},x}| = 0,1 \frac{\text{m}}{\text{s}}$, während der

eigene Tracker bei konstanteren Geschwindigkeitsverläufen bei unter $|\Delta v_{\text{rel},x}| = 0,5 \frac{\text{m}}{\text{s}}$ liegt und variiert bis zu etwa $|\Delta v_{\text{rel},x}| = 0,8 \frac{\text{m}}{\text{s}}$ groß wird. Der Fehler in der Geschwindigkeitsschätzung des eigenen Trackers wird durch Ungenauigkeiten in der Berechnung der kartesischen Geschwindigkeiten und der anschließenden Filterung der prädizierten Zustände erzeugt.

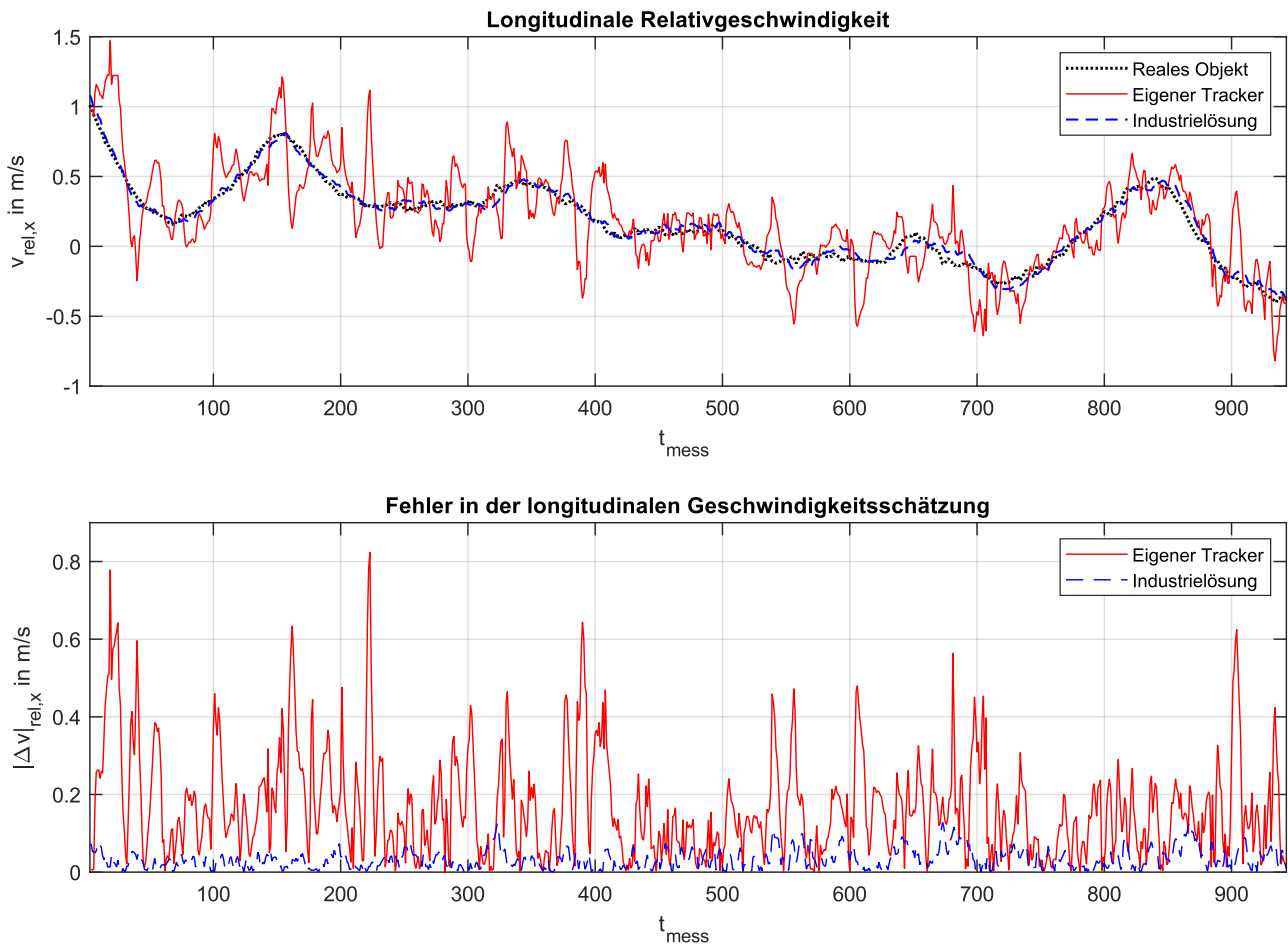


Abbildung 7-3: Vergleich der geschätzten Relativgeschwindigkeiten zwischen implementierten Tracker und der Industrielösung für Szenario 1

Die Ergebnisse einer beschleunigten Fahrt sind nachfolgend zu finden. In Abbildung 7-4 sind die geschätzten Relativgeschwindigkeiten von Messdaten mit einer mittleren Beschleunigung dargestellt. Am Verlauf der realen Geschwindigkeit ist zu erkennen, dass das Zielfahrzeug zuerst von etwa $v_{\text{rel},x} = 6 \frac{\text{m}}{\text{s}}$ auf $v_{\text{rel},x} = -6 \frac{\text{m}}{\text{s}}$ abbremst und anschließend erneut auf $v_{\text{rel},x} = 6 \frac{\text{m}}{\text{s}}$ beschleunigt bis es anschließend eine konstante Relativgeschwindigkeit von etwa $v_{\text{rel},x} = 0$ aufweist. Dieses Szenario verdeutlicht die unterschiedlichen Schätzungen bei beschleunigter Bewegung zwischen der Industrielösung und dem eigenen Tracker. Die Industrielösung weist bei der Geschwindigkeitsschätzung einer beschleunigten Bewegung ($t_{\text{mess}} = 0$ bis $t_{\text{mess}} = 600$) nicht die gleiche Genauigkeit auf wie bei einer konstanten Geschwindigkeit ($t_{\text{mess}} = 600$ bis $t_{\text{mess}} = 1000$). Die Unterschiede zwischen beiden Trackern ist damit bei beschleunigter Bewegung kleiner als bei einer konstanten Geschwindigkeit, wie in Abbildung 7-4 unten zu sehen ist. Die Schwächen der Industrielösung bei

beschleunigter Bewegung werden insbesondere daran deutlich, dass das verfolgte Zielfahrzeug nicht kontinuierlich verfolgt wird, sondern an der Stelle der größten Beschleunigung kurzzeitig verloren geht und einen Fehler von bis zu $|\Delta v_{\text{rel},x}| = 3,3 \frac{\text{m}}{\text{s}}$ erzeugt (rote Kreismarkierungen). Die Fehler während der beschleunigten Bewegung werden bei beiden Trackern durch die Trägheit des Filters verursacht. Diese Trägheit erzeugt einen zeitlichen Verzug in der Geschwindigkeitsschätzung, welche insbesondere zwischen den Zyklen $t_{\text{mess}} = 250$ und $t_{\text{mess}} = 500$ zu erkennen ist.

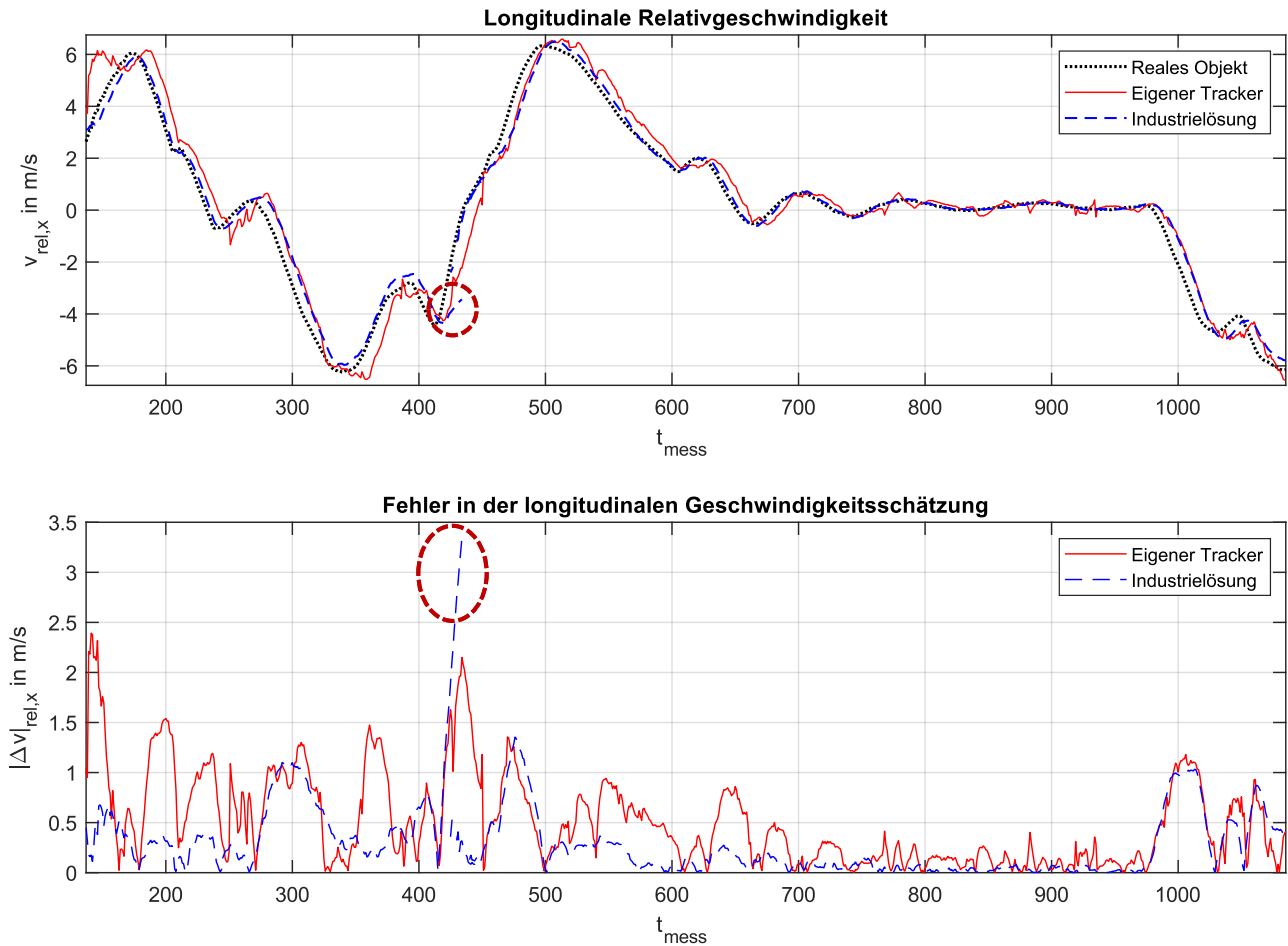


Abbildung 7-4: Vergleich der geschätzten Relativgeschwindigkeiten zwischen implementierten Tracker und der Industrielösung für Szenario 2

Dieser Trägheitseffekt ist für das Szenario 3 in Abbildung 7-5 (oben) deutlicher zu sehen. Das Zielfahrzeug beschleunigt aus dem Stand auf etwa $v_{\text{rel},x} = 33 \frac{\text{m}}{\text{s}}$ mit einer Beschleunigung von bis zu $a_{\text{rel},x} = 5 \frac{\text{m}}{\text{s}^2}$. Durch die monoton ansteigende Geschwindigkeit insbesondere zwischen den Zyklen $t_{\text{mess}} = 40$ und $t_{\text{mess}} = 90$ schwankt der Fehler der Schätzung beider Tracker weniger als im Vergleich zur steigenden und fallenden Geschwindigkeit der vorherigen Szenarien. Dabei steigt in diesem Bereich der Fehler des eigenen Trackers auf etwa $|\Delta v_{\text{rel},x}| = 2,7 \frac{\text{m}}{\text{s}}$, während der Fehler der Industrielösung nur auf etwa $|\Delta v_{\text{rel},x}| = 1,3 \frac{\text{m}}{\text{s}}$ steigt. Dies bedeutet, dass die Dynamik des eigenen Trackers auf Geschwindigkeitsänderungen zu reagieren geringer ist als die der Industrielösung.

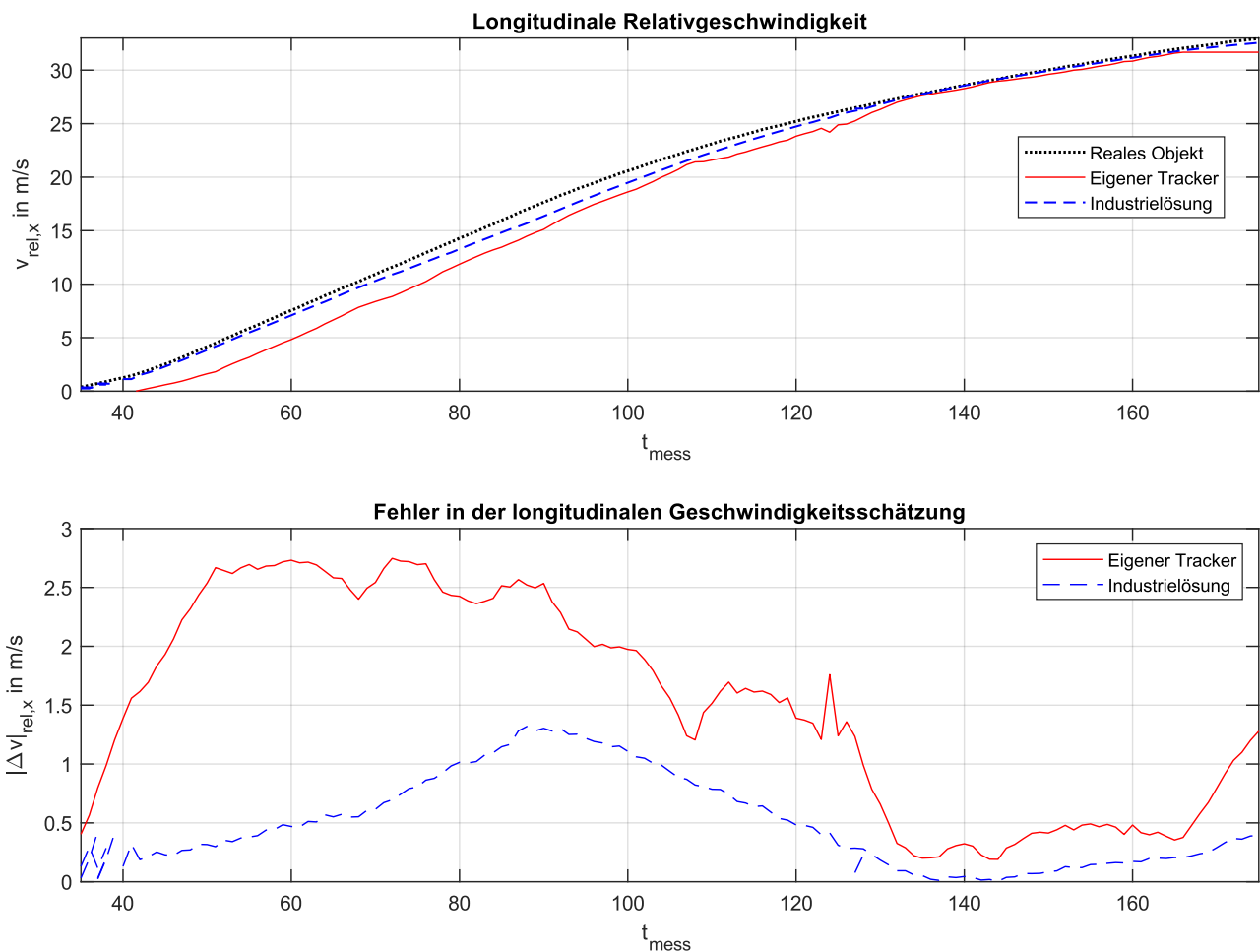


Abbildung 7-5: Vergleich der geschätzten Relativgeschwindigkeiten zwischen implementierten Tracker und der Industrielösung für Szenario 3

Dieser Trägheitseffekt des eigenen Trackers lässt sich ebenfalls in Abbildung 7-6 anhand der größeren Fehler, welche in Abhängigkeit der relativen longitudinalen Beschleunigung aufgetragen sind, erkennen. Zusätzlich wird deutlich, dass für beide Tracker der Fehler der Schätzung mit noch höherer Beschleunigung wächst. Dazu wurden in Abbildung 7-6 jeweils eine Regressionsgerade eingefügt, um die unterschiedlichen Steigungen der Fehler je nach Tracker zu verdeutlichen.

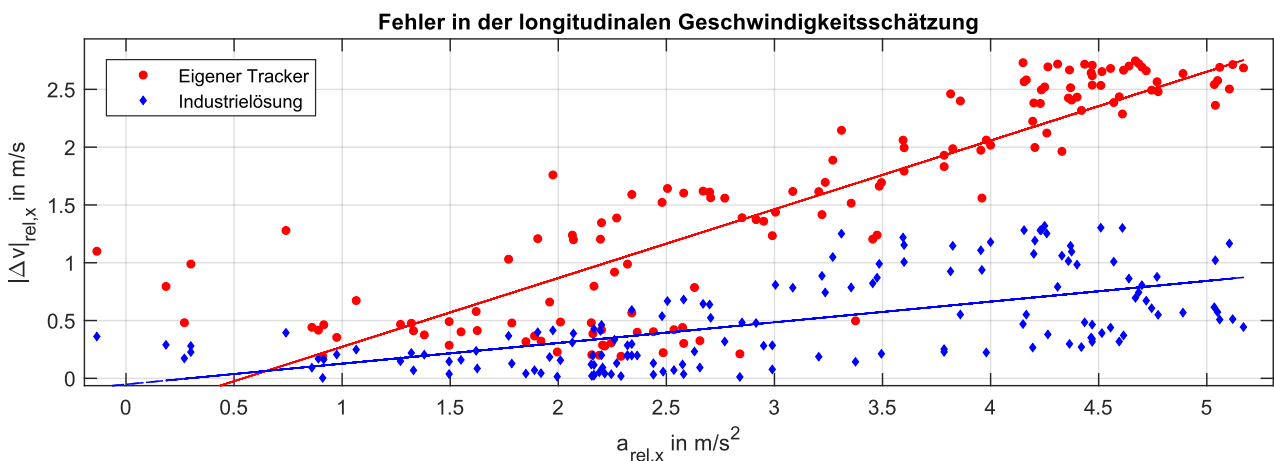


Abbildung 7-6: Fehler in der longitudinalen Geschwindigkeitsschätzung in Abhängigkeit der Beschleunigung für Szenario 3

Analog zum Vergleich der longitudinalen Relativgeschwindigkeiten folgt die Gegenüberstellung der lateralen Geschwindigkeitsschätzungen. In Abbildung 7-7 sind die Schätzungen beider Tracker der lateralen Relativgeschwindigkeit $v_{\text{rel},y}$ sowie die erzeugten Fehler $|\Delta v_{\text{rel},y}|$ über den zeitlichen Verlauf für das Szenario 2 dargestellt. Da das Verhalten der Geschwindigkeitsschätzung für die Szenarien 1 und 3 keine neuen Informationen liefert, wird auf die Darstellungen der zugehörigen Diagramme an dieser Stelle verzichtet. Der unteren Abbildung ist zu entnehmen, dass, während das Zielfahrzeug sich durchgehend mit etwa $v_{\text{rel},y} = 0$ fortbewegt, die Schätzungen beider Tracker von diesem Wert abweichen. Die Industrielösung erreicht dabei einen Fehler von bis zu $|\Delta v_{\text{rel},y}| = 1 \frac{\text{m}}{\text{s}}$, während der Fehler des eigenen Trackers höher liegt und bis zu $|\Delta v_{\text{rel},y}| = 2,5 \frac{\text{m}}{\text{s}}$ groß wird. Geht das verfolgte Objekt verloren ($t_{\text{mess}} = 425$) kann der Fehler bis zu jeweils $|\Delta v_{\text{rel},y}| = 4,5 \frac{\text{m}}{\text{s}}$ und $|\Delta v_{\text{rel},y}| = 3,5 \frac{\text{m}}{\text{s}}$ groß werden. Allerdings wird vorher schon ein neues Objekt initiiert, sodass dieser große Fehler aufgrund der fortgeführten Prädiktion eines Objekts in der Löschphase entsteht. Die genaue Methode der Industrielösung zur Berechnung der kartesischen Geschwindigkeit $v_{\text{rel},y}$ ist nicht bekannt, jedoch fällt auf, dass die Fehler beider Tracker annähernd zu den gleichen Zeitpunkten wachsen, wie in den Zyklen $t_{\text{mess}} = 700$ bis $t_{\text{mess}} = 1000$, obwohl die reale Geschwindigkeit des Objekts nicht stark variiert.

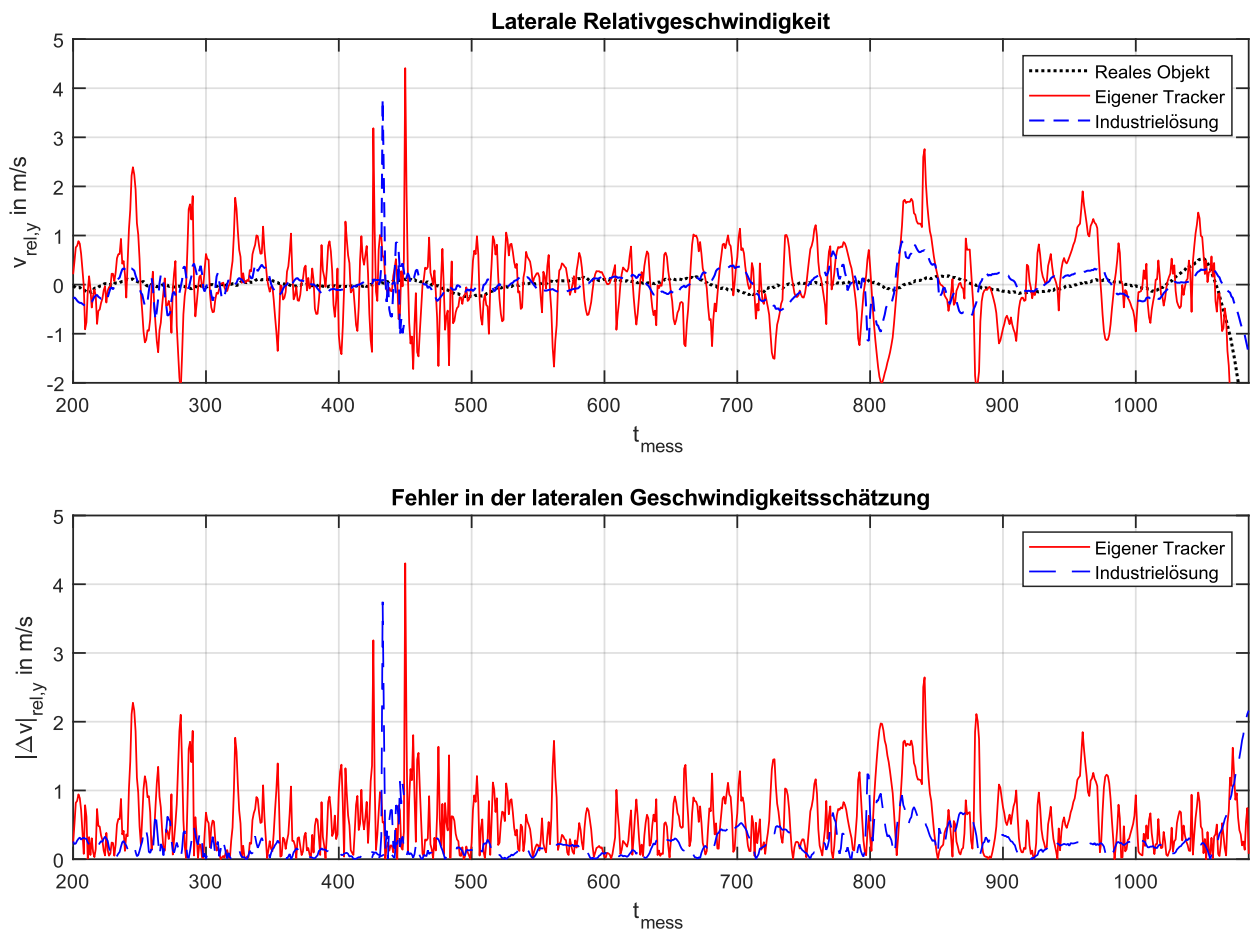


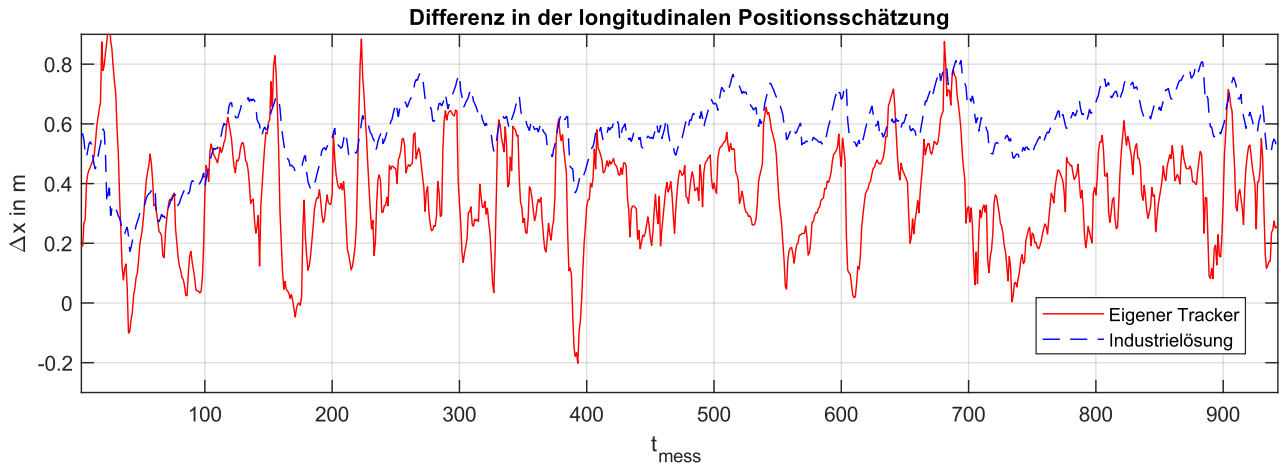
Abbildung 7-7: Vergleich der geschätzten lateralen Relativgeschwindigkeiten zwischen implementierten Tracker und der Industrielösung für Szenario 2

7.2.3 Positionsschätzung

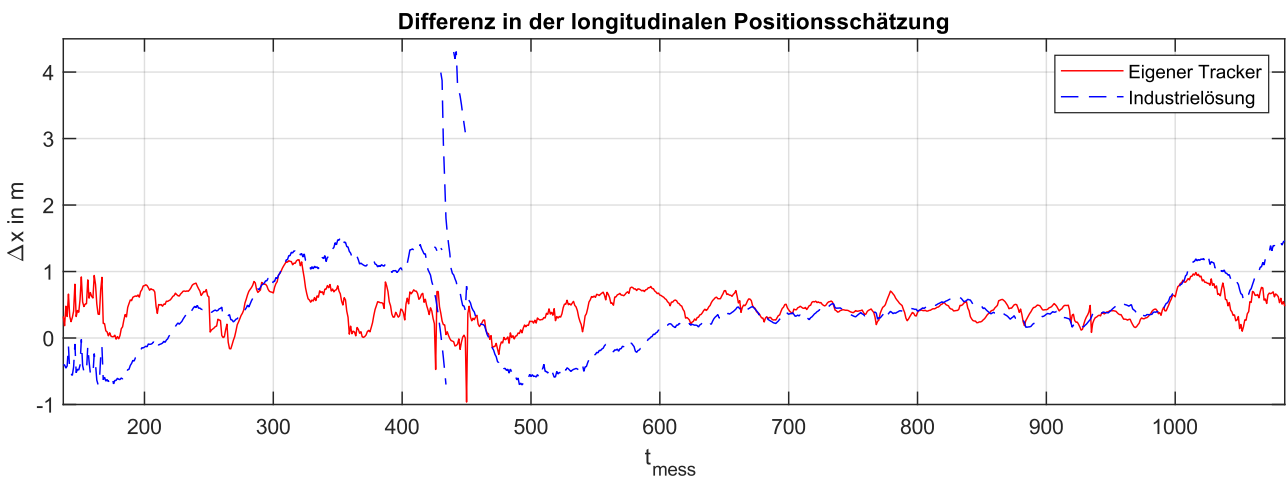
Die Fehler in der longitudinalen Geschwindigkeitsschätzung führen zu Ungenauigkeiten in der Positionsschätzung der Zielfahrzeuge. Die Differenz wurde jeweils wie folgt berechnet und bezieht sich jeweils auf den nächsten Punkt der Positionsschätzung zum Egofahrzeug (in diesen Szenarien ist dies das Fahrzeugheck)

$$\Delta x = x_{\text{track}} - x_{\text{real}} \quad (7-1)$$

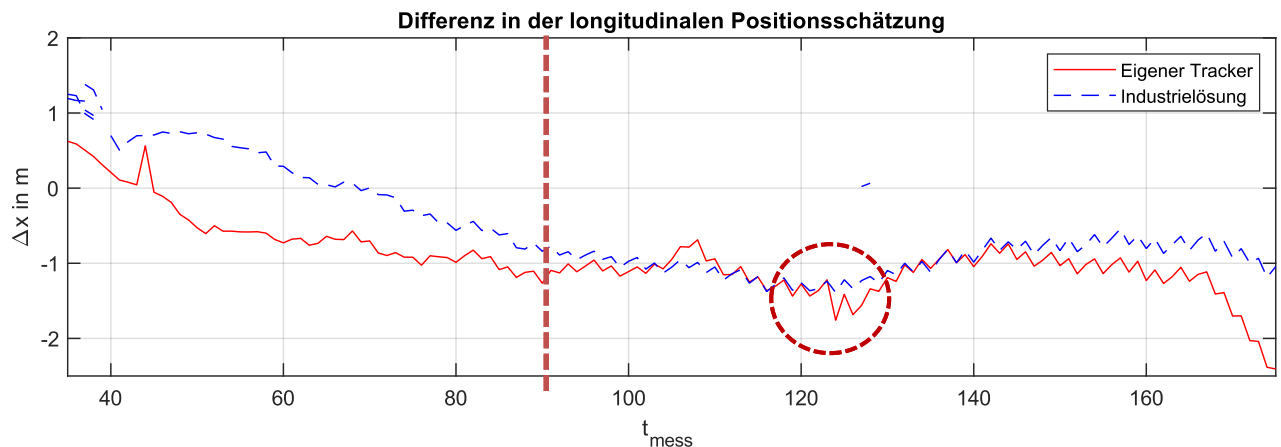
wobei x_{track} die geschätzte longitudinale Position im Tracker und x_{real} die reale Position darstellt. Diese Größe wird vorzeichenbehaftet dargestellt, um zu erkennen, ob der jeweilige Tracker das Zielfahrzeug näher oder entfernter vom Egofahrzeug schätzt. Im folgenden Abschnitt sind die Szenarien 1-3 beschrieben, welche ebenfalls im vorherigen Kapitel der Geschwindigkeitsschätzung beschrieben und ausgewertet wurden. Aus Abbildung 7-8 (a) für das Szenario 1 ist zu erkennen, dass trotz des höheren Fehlers in der Geschwindigkeitsschätzung durch den eigenen Tracker, dieser nicht zwingend eine höhere Differenz in der Positionsschätzung generiert. Die Industrielösung schätzt das Zielfahrzeug bei einer annähernd konstanten Geschwindigkeit grundlegend mehr als $\Delta x = 0,4$ m zu weit, während die Schätzung des eigenen Trackers genauer ist und hier zwischen $\Delta x = 0$ und $\Delta x = 0,6$ m abweicht, und vereinzelt auf bis zu $\Delta x = 0,9$ m steigt. Bei leicht beschleunigten Bewegungen, wie in Szenario 2, schwanken die Werte der Positionsschätzungen beider Tracker mehr, sodass höhere Differenzen erzeugt werden, wie in Abbildung 7-8 (b) zu sehen ist. Während die Differenz der Positionsschätzung durch den eigenen Tracker zwischen $\Delta x = 0$ und $\Delta x = 1$ m liegt, schwankt die Differenz der Industrielösung zwischen $\Delta x = -0,75$ m und $\Delta x = 1,5$ m. Die Ausschläge aufgrund der hohen Beschleunigung bei $t_{\text{mess}} = 425$ reichen jeweils bis zu $\Delta x = -1$ m für den eigenen Tracker und $\Delta x = 4$ m für die Industrielösung. Dabei schätzt der Tracker der Industrielösung das Zielfahrzeug entfernter, wenn es sich annähert, und näher, wenn es sich entfernt. Hier ist erneut ein Trägheitseffekt zu erkennen, sodass die größte Differenz in der Positionsschätzung zeitlich versetzt zum größten Fehler in der Geschwindigkeitsschätzung auftritt. Dieser Effekt ist in Abbildung 7-8 (c) für das Szenario 3 deutlich zu erkennen. Der Zyklus mit dem größten Fehler in der Positionsschätzung ist mithilfe der roten Markierung (Linie) hervorgehoben und liegt bei etwa $t_{\text{mess}} = 90$. Der größte Fehler in der Positionsschätzung tritt allerdings versetzt erst kurz nach $t_{\text{mess}} = 120$ auf (roter Kreis) und liegt für beide Tracker bei ca. $\Delta x = -1,5$ m – das Zielfahrzeug wird zu nah geschätzt. Damit ist festzulegen, dass der Fehler in der Positionsschätzung mit dem Fehler in der Geschwindigkeitsschätzung mit einem zeitlichen Verzug steigt. Allerdings unterscheidet sich der Einfluss der Fehler in der Geschwindigkeit auf die Positionsschätzung je nach Tracker, da im Vergleich ein höherer Fehler des einen Trackers in der Geschwindigkeit nicht zwingend einen höheren Fehler in der Position bedeutet.



(a): Differenz in der longitudinalen Positionsschätzung für Szenario 1



(b): Differenz in der longitudinalen Positionsschätzung für Szenario 2

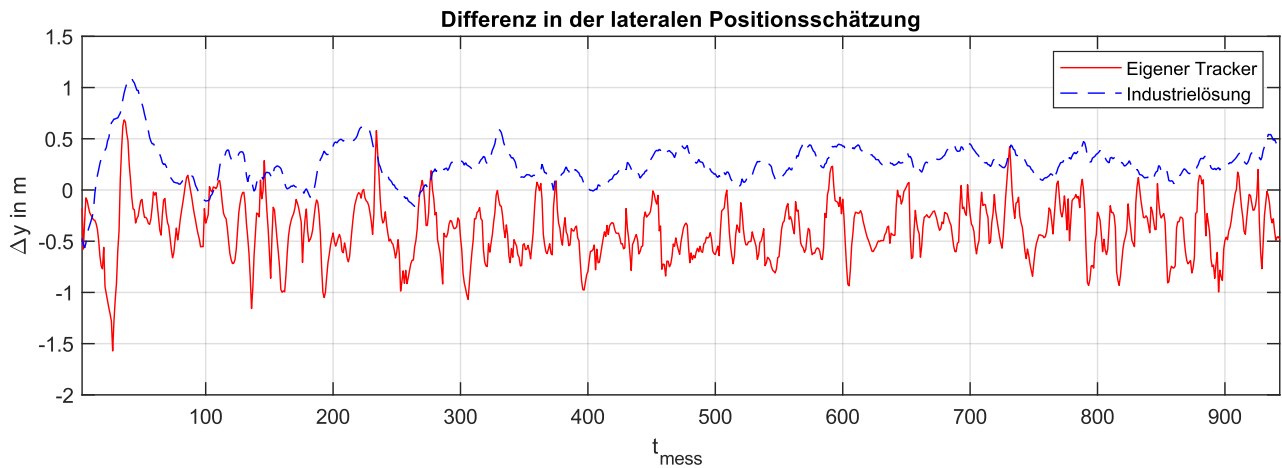


(c): Differenz in der longitudinalen Positionsschätzung für Szenario 3

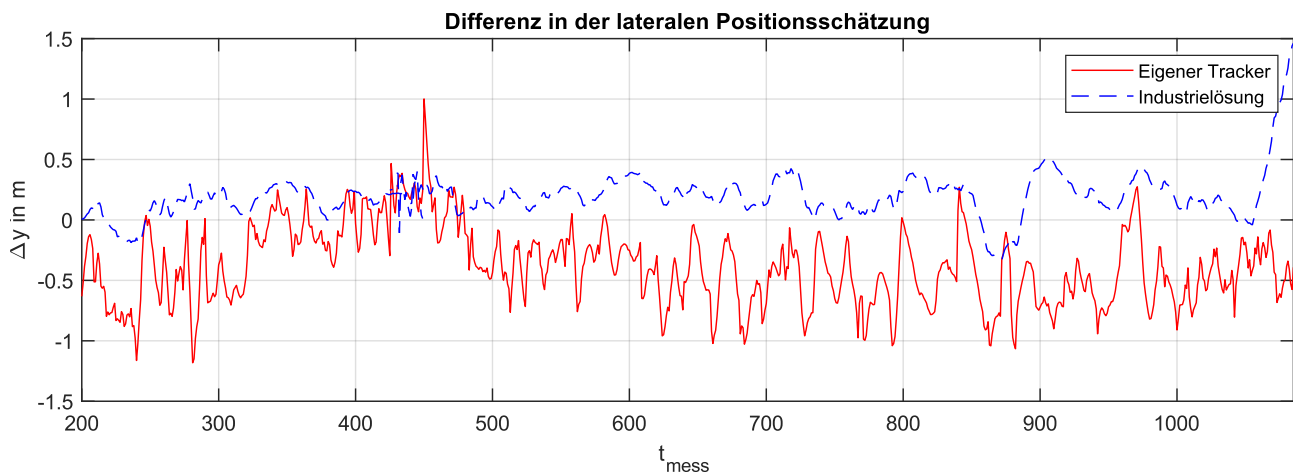
Abbildung 7-8: Vergleich der Differenzen in der longitudinalen Positionsschätzung für Szenario 1 bis 3

Zusätzlich zur longitudinalen Positionsschätzung sind in Abbildung 7-9 die Differenzen der lateralen Positionsschätzung für die Szenarien 1 bis 3 dargestellt. Die Referenzpunkte beziehen sich jeweils auf den Mittelpunkt der Fahrzeuglängsachse und werden analog zu (7-1) berechnet. Die Ergebnisse der Positionsschätzung der Industrielösung sind für alle drei Szenarien nahezu identisch. Die erzeugte

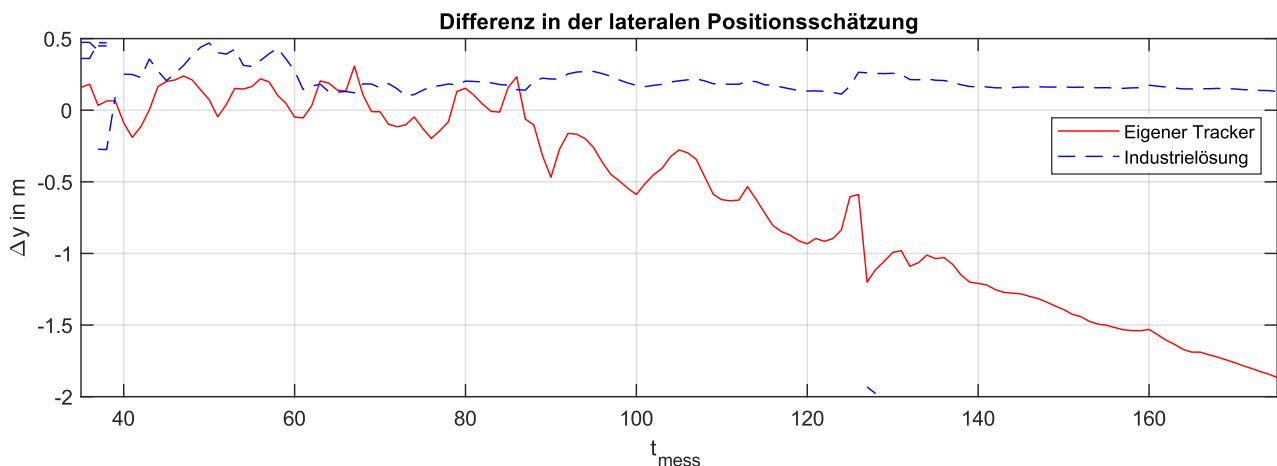
Differenz liegt grundlegend zwischen $\Delta y = 0$ und $\Delta y = 0,5$ m – vereinzelt werden größere Fehler erzeugt, deren Ursache nicht bekannt ist. Die Differenzen des eigenen Trackers sind betragsmäßig größer und liegen zwischen $\Delta y = -2$ m und $\Delta y = 1$ m für alle drei Szenarien. Zu erkennen ist hier, wie der Fehler mit der Beschleunigung des Zielfahrzeuges sinkt, welches in den Zyklen $t_{\text{mess}} = 40$ $t_{\text{mess}} = 80$ in Abbildung 7-9 (c) zu erkennen ist. Im Mittel beträgt die Differenz hier $\Delta y = 0$, während die Differenz einer annähernd konstanten Fahrt in Abbildung 7-9 (a) und (b) im Mittel bei etwa $\Delta y = -0,5$ liegt. Der steigende Fehler in Abbildung 7-9 (c) zwischen den Zyklen $t_{\text{mess}} = 80$ und $t_{\text{mess}} = 180$ ist vielmehr auf die steigende Entfernung zurückzuführen als auf die Abnahme der Beschleunigung. Die Unterschiede in der Positionsschätzung beider Tracker ist zum einen auf Ungenauigkeiten in der Geschwindigkeitsschätzung zurückzuführen, aber auch auf die Schätzung der Bounding-Box. Diese ist insbesondere, bei wenigen Detektionen für ein Objekt, deren Verteilung nur ungenügend Informationen über die Objektdimensionen liefert, instabil.



(a): Differenz in der lateralen Positionsschätzung für Szenario 1



(b): Differenz in der lateralen Positionsschätzung für Szenario 2



(c): Differenz in der lateralen Positionsschätzung für Szenario 3

Abbildung 7-9: Vergleich der Differenzen in der lateralen Positionsschätzung für Szenario 1 bis 3

Die Bewertung der Positionsschätzung beider Tracker erfolgte zusätzlich mit einer Kurvenfahrt. Hier ist für die Fahrstreifenzuweisung des Zielfahrzeugs in einer möglichen Trajektorienplanung des Egofahrzeugs neben der longitudinalen auch die laterale Schätzung ausschlaggebend. Beide Richtungen der Schätzungen werden damit in einem Beispiel vereint. In dem ausgewählten Szenario fährt das Zielfahrzeug bei $x = 55$ m mit $v = 30 \frac{\text{m}}{\text{s}}$ in negative y-Richtung des Fahrzeugkoordinatensystems in das Sichtfeld des Sensors ein und macht anschließend eine Rechtskurve, sodass es in Richtung der negativen x-Achse auf das Egofahrzeug zufährt. Dieses Szenario wurde dreimal durchgeführt und aufgenommen. Die Trajektorien des Zielfahrzeuges (Reales Objekt) sind in Abbildung 7-10 dargestellt. Zusätzlich sind die Verläufe der Positionsschätzungen der drei Durchläufe sowie die Fahrbahn mit zwei Fahrstreifen abgebildet. Das Zielfahrzeug befindet sich auf dem inneren Fahrstreifen der Kurve.

Betrachtet man den gesamten Verlauf der Positionsschätzungen, ist zu sehen, dass die Industrielösung einen glatteren Verlauf ausgibt, während die Schätzung des eigenen Trackers stärker schwankt. Im ersten Abschnitt der Trajektorie des realen Objekts von $x = 55$ m bis etwa $x = 50$ m ist für beide Tracker zu sehen, dass die Systeme den Eintrittspunkt des Zielfahrzeugs in das Sensorsichtfeld fehlerbehaftet schätzen. Der eigene Tracker weist dabei eine Differenz von etwa $\Delta y = -2,7$ m auf, während die Industrielösung bei nur etwa $\Delta y = -1$ m liegt. Allerdings verfolgt dieser Tracker für zwei Durchläufe zwei Objekte für das gleiche Zielfahrzeug, welches an der Zahl der blauen Graphen im Eintrittsbereich von $x = 55$ m bis etwa $x = 50$ m zu erkennen ist. Beide Objekte werden kurzzeitig verfolgt, bis bei etwa $x = 50$ m eins dieser Objekte gelöscht wird. Die Bedingungen für eine Objektlöschung im Tracker der Industrielösung sind nicht bekannt. In diesem ersten Abschnitt ist die Positionsschätzung der Industrielösung genauer, während der eigene Tracker mit dem genannten Fehler das Zielobjekt teilweise auf dem Egofahrstreifen annimmt.

Im anschließenden zweiten Abschnitt nähert sich die Positionsschätzung des eigenen Trackers dem realen Trajektorienverlauf an, sodass die Differenz der lateralen Schätzung im Mittel etwa $\Delta y = 0,5$ m beträgt. Die Schätzung der Industrielösung hingegen weist für zwei Durchläufe bis etwa $x =$

30 m eine höhere Differenz von etwa $\Delta y = -1,5$ m auf, sodass bei einer Fahrstreifenbreite von 3,5 m Teile des Fahrzeuges auf dem eigenen Fahrstreifen angenommen werden. Diese Schätzung wird im weiteren Verlauf genauer und nähert sich ebenfalls einer Differenz von $\Delta y = 0,5$ m an. Abschließend lässt sich zu diesem Kurvenszenario sagen, dass trotz des kleineren Fehlers in der lateralen Positionsschätzung der vorherigen Szenarien, dieser Fehler nicht notwendigerweise auch in Szenarien, in denen sich die reale laterale Entfernung ändert, kleiner ist.

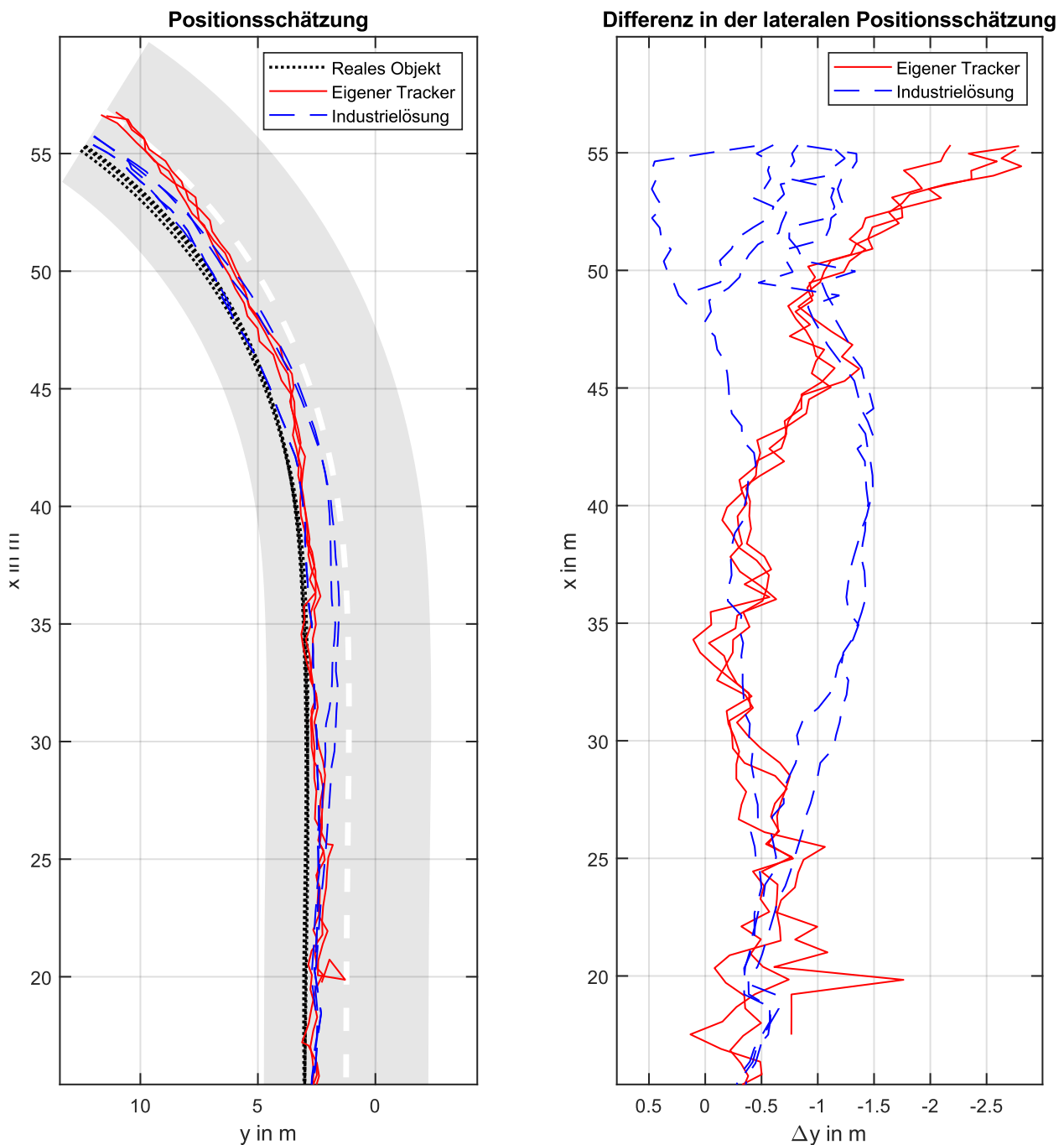


Abbildung 7-10: Vergleich der Positionsschätzung für eine Kurvenfahrt und Darstellung der lateralen Differenz

7.2.4 Gierwinkelschätzung

Der geschätzte Gierwinkel ergibt sich im eigenen Tracker aus beiden kartesischen Geschwindigkeiten. Für diesen Vergleich wird eine Slalomfahrt des Zielfahrzeuges gewählt. Das Zielfahrzeug bewegt sich mit etwa $v_{\text{rel}} = 11 \frac{\text{m}}{\text{s}}$, wobei die Slalomtrajektorie sich über die gesamte Fahrbahnbreite von 20 m erstreckt. In Abbildung 7-11 sind im oberen Diagramm der Gierwinkel des realen Objekts sowie die geschätzten Gierwinkel der Industrielösung und des eigenen Trackers im Verlauf des Szenarios abgebildet. Zusätzlich finden sich im unteren Diagramm die Differenzen beider Tracker. Man erkennt analog zur Geschwindigkeits- und Positionsschätzung einen Unterschied in der Stabilität der Schätzung beider Tracker. Die Industrielösung weist einen deutlich glatteren Verlauf der Schätzung auf, während der eigene Tracker kontinuierlich innerhalb eines Bereichs von etwa 10° um den realen Wert schwankt. Die erzeugten Differenzen unterscheiden sich damit ebenfalls. Die Differenzen zeigen aufgrund der Trägheit beider Systeme einen periodischen Verlauf zwischen $\Delta\varphi = \pm 20^\circ$ für den Tracker der Industrielösung und zwischen $\varphi = \pm 10^\circ$ für den eigenen Tracker. Dieser Fehler wird durch eine Phasenverschiebung der geschätzten Gierwinkel zum realen Wert erzeugt, wie im oberen Diagramm für beide Systeme zu erkennen ist. Diese Phasenverschiebung beträgt für die Industrielösung durchschnittlich etwa $\Delta t_{\text{mess}} = 13$ und für den eigenen Tracker etwa $\Delta t_{\text{mess}} = 5$. Zusätzlich erzeugt die Trägheit der Industrielösung nicht nur eine Phasenverschiebung, sondern auch eine Erhöhung der Amplitude des geschätzten Gierwinkels um etwa 5° . Dieser Effekt ist beim eigenen Tracker nicht zu beobachten. Im oberen sowie im unteren Diagramm ist für die Industrielösung in jeder Periode eine zweite Linie vorhanden, welche beispielhaft mit der roten Kreismarkierung hervorgehoben ist. Diese zweite Linie deutet auf ein zweites initiiertes Objekt, sodass simultan zwei Objekte verfolgt werden bei einem Zielfahrzeug. Eine nicht erfolgreiche Datenassoziation zum ersten Objekt kann der Grund hierfür sein. Das zweite initiierte Objekt wird jedoch kurz drauf wieder gelöscht, sodass nur noch das ursprünglich generierte Objekt vorhanden ist. Dieses zweite Objekt wird stets am Punkt der größten Gierrate und damit der größten Differenz des geschätzten Gierwinkels des ersten Objekts initiiert.

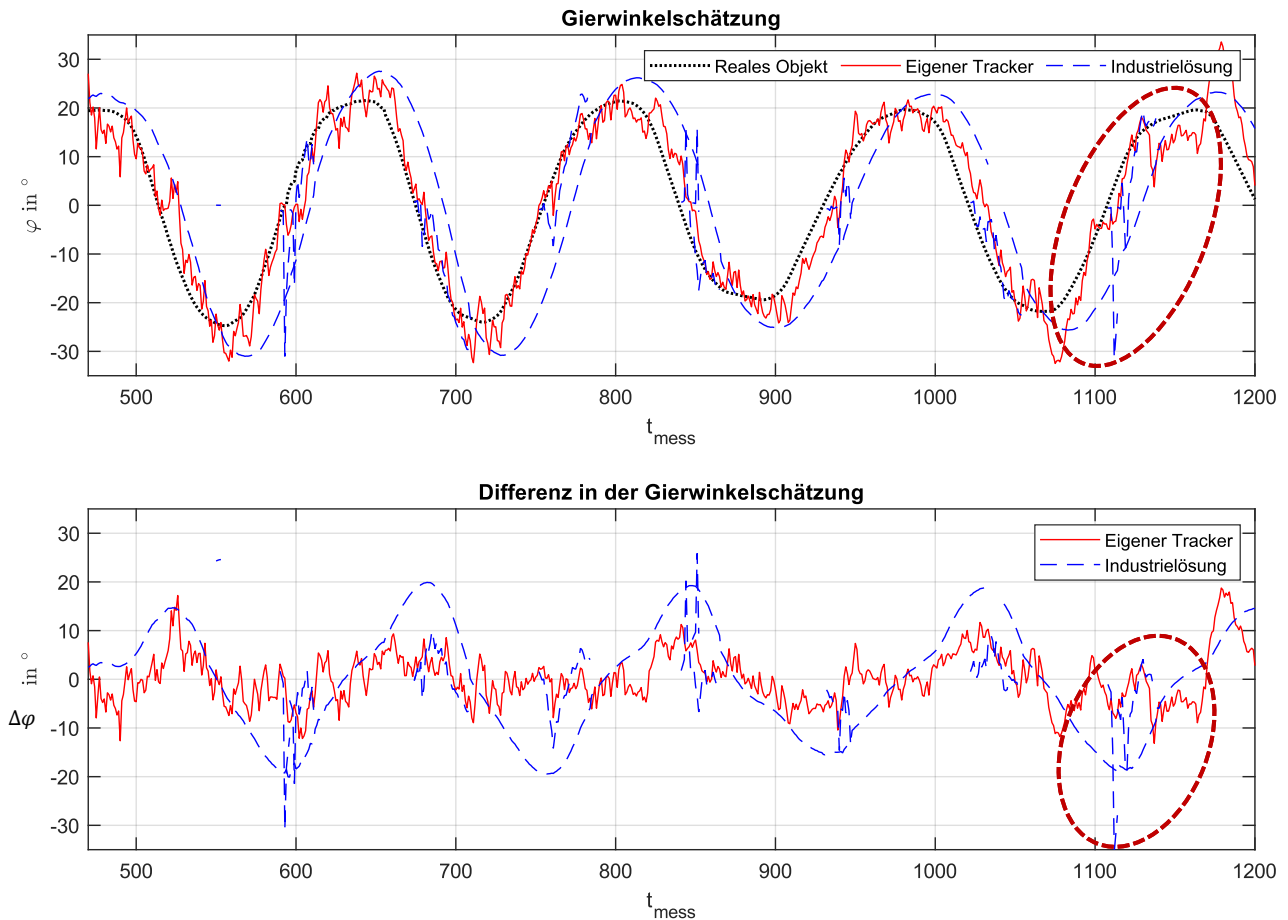


Abbildung 7-11: Vergleich des geschätzten Gierwinkels φ zwischen implementierten Tracker und der Industrielösung für eine Slalomfahrt

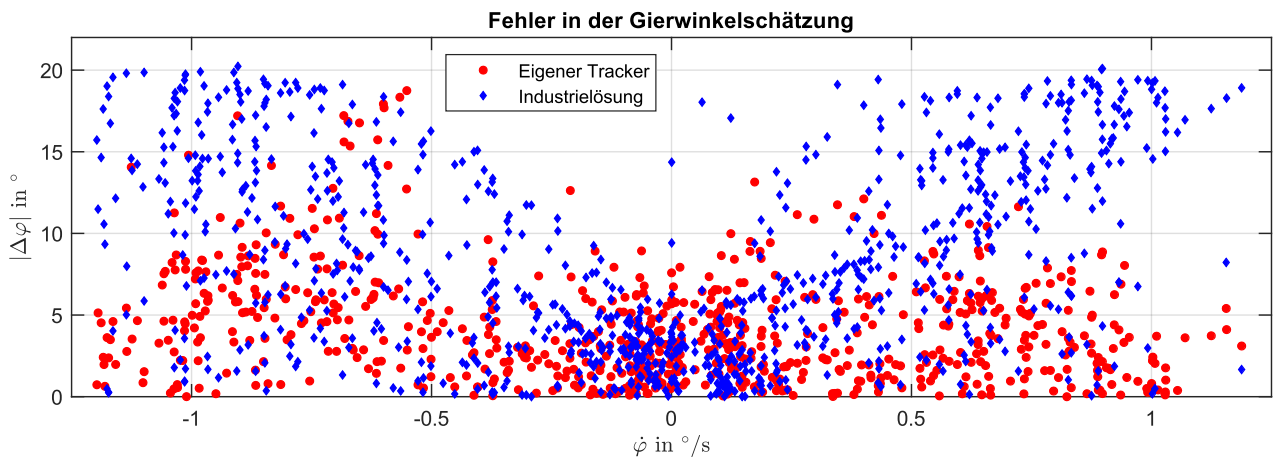


Abbildung 7-12: Vergleich der Fehler in der Gierwinkelschätzung $|\Delta\varphi|$ in Abhängigkeit der Gierrate $\dot{\varphi}$

Dieser Zusammenhang ist ebenfalls in Abbildung 7-12 zu erkennen, in der der Fehler in der Gierwinkelschätzung $|\Delta\varphi|$ in Abhängigkeit der Gierrate $\dot{\varphi}$ aufgetragen ist. Für beide Tracker lässt sich ein höherer Fehler mit höherer Gierrate erkennen, wobei für die Industrielösung dieser Effekt aufgrund der Trägheit deutlich ausgeprägter ist.

7.2.5 Bounding-Box-Schätzung

Die Bounding-Box eines Objekts ist abhängig von der Position und des Gierwinkels des Fahrzeuges sowie von den bekannten Fahrzeugdimensionen. Um die Schätzung der Bounding-Box beider Tracker zu vergleichen wird ein Szenario mit drei Fahrstreifen verwendet, bei dem die äußeren Fahrstreifen durch je ein statisches Fahrzeug belegt ist. Damit sind zwei Zielfahrzeuge vorhanden. Das Egofahrzeug nähert sich auf dem mittleren Fahrstreifen den beiden Zielfahrzeugen, ähnlich einer Ampelsituation. Das Ziel dieses Szenarios ist die Bounding-Box des jeweiligen Fahrzeuges so genau zu schätzen, dass nur die äußeren Fahrstreifen als belegt und der Egofahrstreifen als vollständig frei angenommen wird. Die Positionsschätzung beider Zielfahrzeuge relativ zum jeweiligen Fahrstreifen ist in Abbildung 7-13 dargestellt. Da sich nur das Egofahrzeug bewegt, schwankt dessen relative Position zum mittleren Fahrstreifen sowie zu beiden Zielfahrzeugen. Diese Schwankungen erkennt man am geschwungenen Verlauf der Fahrbahn und der Darstellung der Position der realen Objekte. Der Referenzpunkt der dargestellten geschätzten Position ist die Mitte der Bounding-Box.

In der Abbildung ist deutlich zu sehen, dass die Positionsschätzung der Industrielösung stabiler verläuft als die des eigenen Trackers. Während die Industrielösung die Position stets im zugehörigen Fahrstreifen schätzt, ergeben sich für den eigenen Tracker durch die Schwankungen Schätzungen, welche in die benachbarten Fahrstreifen ausschlagen.

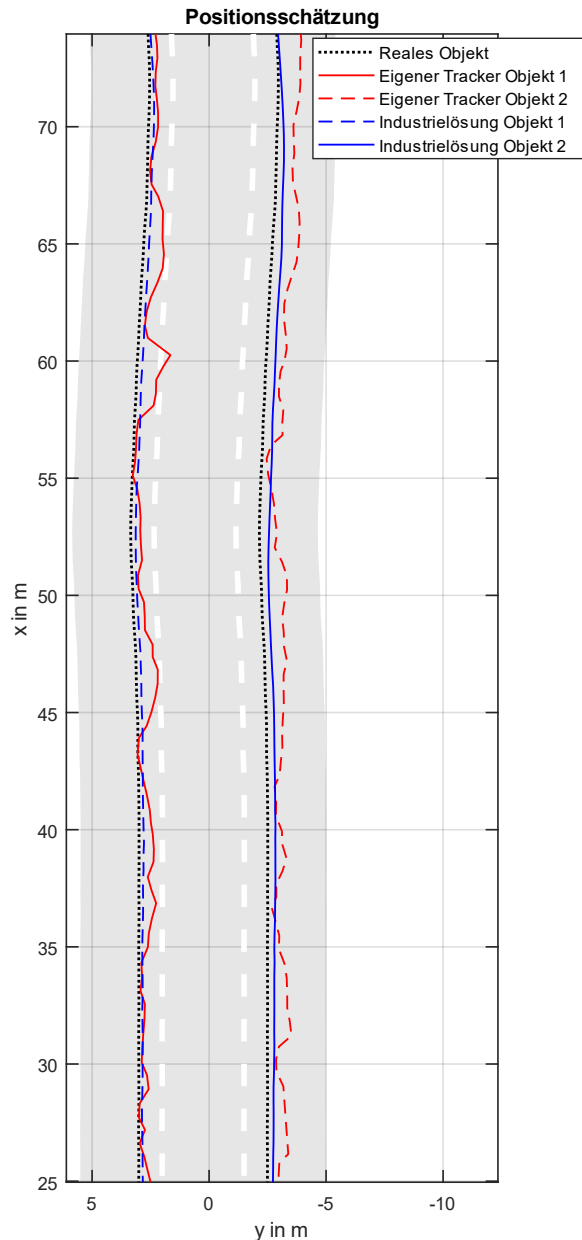


Abbildung 7-13: Schätzung der kartesischen Entfernungen zweier Zielfahrzeuge zur Bewertung der Bounding-Box-Schätzung

Die zugehörigen Gierwinkel für die Objekte sind in Abbildung 7-14 über die relative Entfernung in x -Richtung aufgetragen. Während die Industrielösung den Gierwinkel für statisch klassifizierte Objekte fest auf $\varphi = 0$ setzt, der eigene Tracker vereinzelt Ausschläge bis zu $\varphi = 100^\circ$ aufweist. Der reale Winkel liegt bei $\varphi = -180^\circ$, da sich das Egofahrzeug auf die Zielobjekte zubewegt. Für die Darstellung der Bounding-Box macht es allerdings keinen Unterschied, ob der Winkel bei $\pm 180^\circ$ oder 0° liegt.

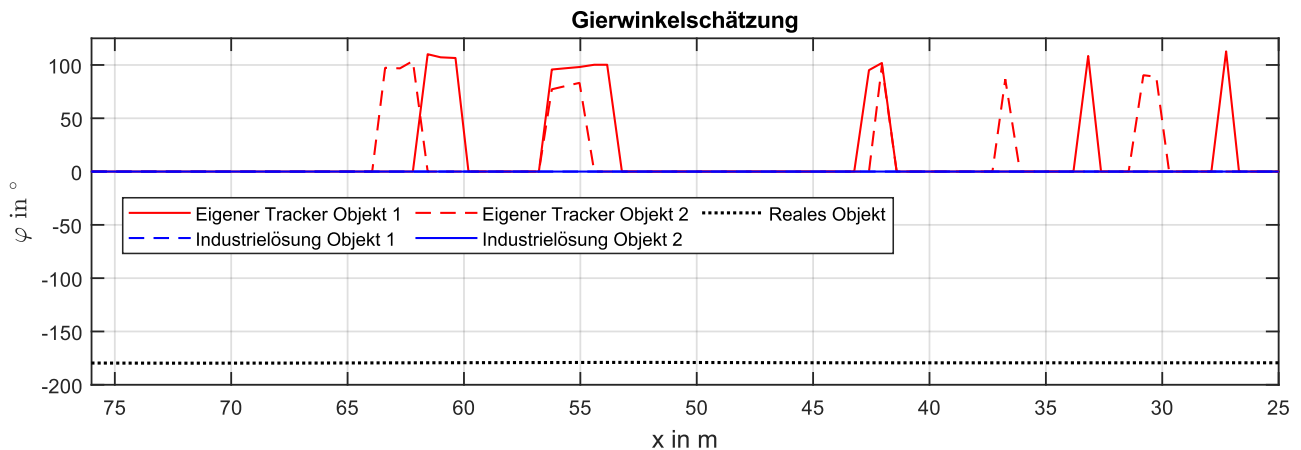


Abbildung 7-14: Vergleich der Gierwinkelschätzung der statischen Zielfahrzeuge aufgetragen über die relative Entfernung in x-Richtung

Aus den errechneten Gierwinkeln, der Positionsschätzungen sowie den geschätzten Objektdimensionen ergeben sich unter Anwendung der in Kapitel 6.6.5 vorgestellten Methode die Schätzungen der Bounding-Boxen, welche für das Szenario beispielhaft in Abbildung 7-15 dargestellt sind. Man erkennt die drei Fahrstreifen sowie zwei gelbe (Industrielösung) und zwei grüne (eigener Tracker) Bounding-Boxen mit ihrer entsprechenden Track-ID. Beide Tracker weisen in diesem Schritt des Tracking-Algorithmus noch Defizite auf. Während die Schätzung der Industrielösung bezüglich der Richtung der Bounding-Box genauer ist, ist die geschätzte Größe der Bounding-Box des eigenen Trackers genauer. Dies liegt daran, dass dieser Tracker eine Mindestgröße der Bounding-Box bzw. der Objekte annimmt, um Objekte mit nur einer Detektion nicht deutlich zu unterdimensionieren. Die ungenauere Bestimmung des Gierwinkels des eigenen Trackers und die damit verbundene Drehung der Bounding-Box hat zur Folge, dass der eigene Fahrstreifen als teilweise belegt angenommen wird und so eine mögliche Trajektorienplanung beeinflusst.

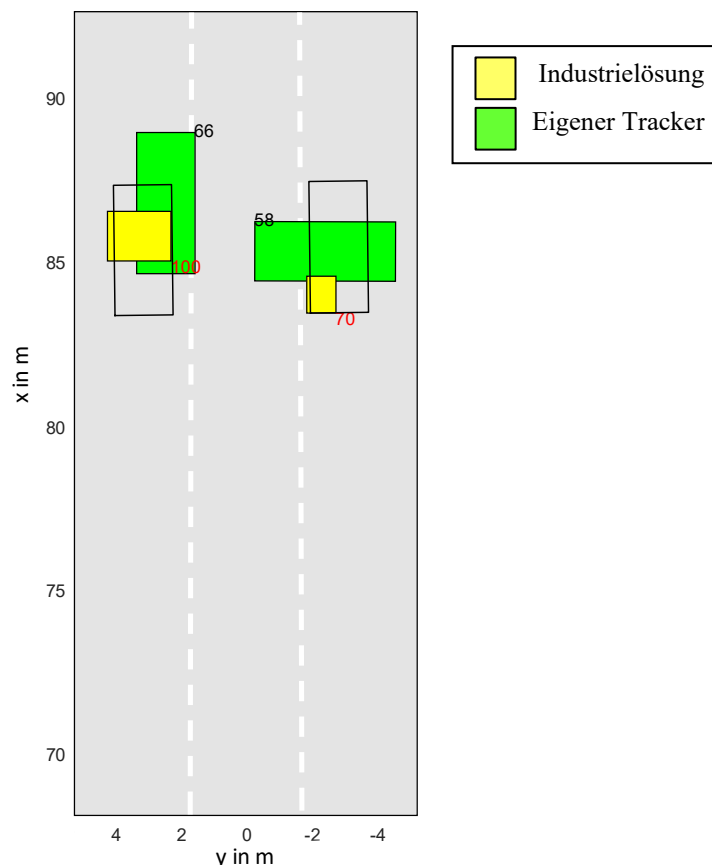


Abbildung 7-15: Beispielhafter Vergleich der geschätzten Bounding-Boxen beider Tracker

7.2.6 Trennfähigkeit

Die Trennfähigkeit der Tracking-Algorithmen von mindestens zwei Objekten wird anhand der im Clustering genutzten Größen, der Position und der Geschwindigkeit, getestet. Mit einer Variation der Grenzwerte zu diesen Parametern wird das Ergebnis der Trennfähigkeit beeinflusst. Im ersten verwendeten Szenario bewegen sich zwei Fahrzeuge mit einer definierten lateralen Distanz und der gleichen Geschwindigkeit vom Egofahrzeug weg, also in positive x-Richtung. Eine laterale Bewegung findet nicht statt. Im Verlauf des Szenarios beschleunigt ein Fahrzeug, während die laterale Distanz nicht variiert. Ziel ist es, bei identischer Geschwindigkeit und geringer Distanz beide Zielfahrzeuge zunächst als nur ein Objekt wahrzunehmen und zu verfolgen und, sobald die Geschwindigkeit des einen Fahrzeuges variiert, jedes Zielfahrzeug als eigenes Objekt zu erfassen. So wird der benötigte Geschwindigkeitsunterschied zwischen zwei Objekten zur getrennten Verfolgung identifiziert. Analog wird das zweite Szenario gestaltet, bei dem die Geschwindigkeit identisch bleibt, während sich die laterale Distanz erhöht. Damit wird die benötigte Distanz zwischen zwei Objekten zur getrennten Verfolgung identifiziert.

In Abbildung 7-16 sieht man die longitudinale Geschwindigkeitsschätzung im zeitlichen Verlauf für das Szenario, welches den nötigen Geschwindigkeitsunterschied zwischen zwei Objekten zur getrennten Verfolgung identifiziert. Damit haben die Objekte eine unveränderte laterale Distanz während des Szenarios. Für beide Tracker ist zu erkennen, dass jeweils nur ein Objekt verfolgt wird

während die Zielfahrzeuge eine annähernd identische Geschwindigkeit besitzen. Bezüglich dieser Größe ist die Trennfähigkeit des eigenen Trackers größer, da dieser zwei Objekte mit einer Geschwindigkeitsdifferenz von mindestens etwa $\Delta v_{\text{rel},x} = 2,7 \frac{\text{m}}{\text{s}}$ getrennt verfolgt, während die Industrielösung eine Differenz von etwa $\Delta v_{\text{rel},x} = 3,2 \frac{\text{m}}{\text{s}}$ benötigt bei identischer lateraler Geschwindigkeit.

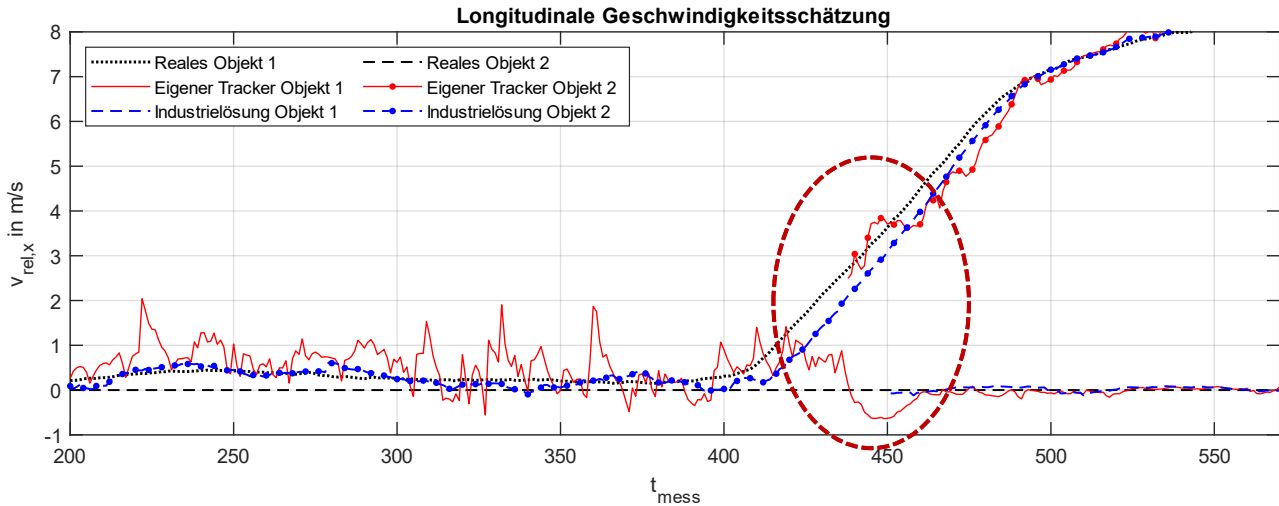


Abbildung 7-16: Longitudinale Geschwindigkeitsschätzung zur Bewertung der Trennfähigkeit der Tracker

In Abbildung 7-17 ist die Positionsschätzung zweier Objekte mit annähernd identischer Geschwindigkeit sowie ihre realen Positionen dargestellt. Die Fahrzeuge haben zu Beginn bei einer Entfernung von $x = 120 \text{ m}$ eine laterale Distanz von $\Delta y = 3 \text{ m}$. Beide fahren in negative x-Richtung auf das Egofahrzeug zu und beginnen ab einer Distanz von etwa $x = 80 \text{ m}$ kontinuierlich auseinanderzudriften. Beide Tracker verfolgen bei $\Delta y = 3 \text{ m}$ mindestens ein Objekt. Für die Industrielösung fällt auf, dass kontinuierlich das Objekt 1 verfolgt wird und vereinzelt ein weiteres Objekt 2. Der Unterschied beider Zielfahrzeuge ist für die Industrielösung demnach vereinzelt groß genug, um beide Objekte getrennt zu verfolgen. Der eigene Tracker hingegen verfolgt bis etwa $x = 55 \text{ m}$ nur ein Objekt. Die Positionsschätzung beider Tracker schwankt hier deutlich mehr als in den Szenarien der vorherigen Kapitel, da Detektionen und damit Informationen zweier Fahrzeuge in einem Objekt zusammengefasst werden. Im markierten Abschnitt (roter Kreis) ist zu erkennen, dass die Industrielösung bei bereits einer kleineren lateralen Distanz ($\Delta y = 4,7 \text{ m}$) zwischen den Zielfahrzeugen zwei Objekte verfolgt als der eigene Tracker ($\Delta y = 6,2 \text{ m}$). Damit ist die Trennfähigkeit der Industrielösung bezüglich dieser Größe größer.

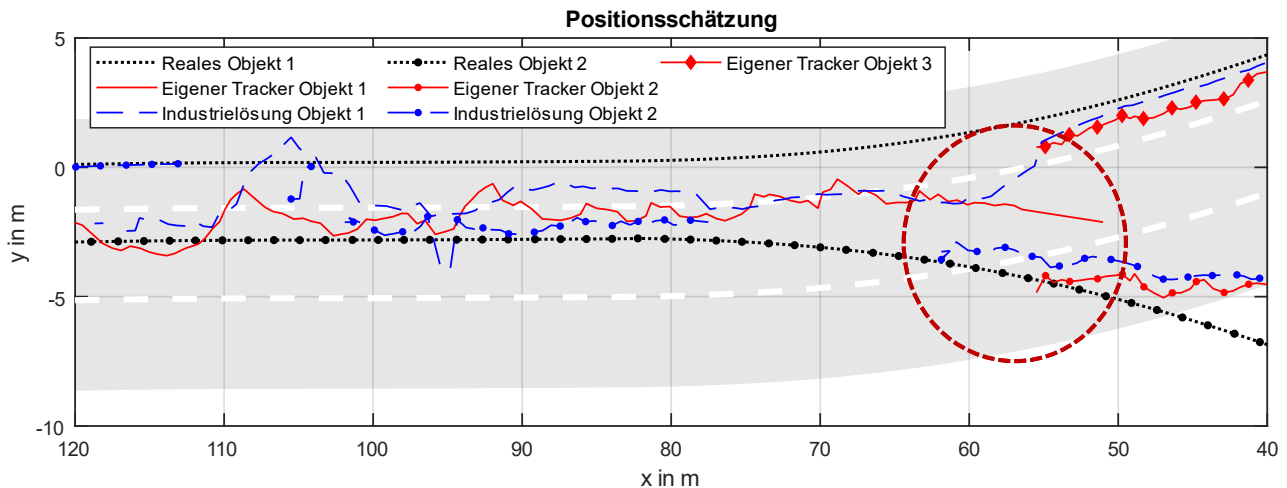


Abbildung 7-17: Positionsschätzung zweier Fahrzeuge zur Bewertung der Trennfähigkeit der Tracker

7.2.7 Kontinuität

Um die Kontinuität beider Tracker zu bewerten wird ein Szenario verwendet, welches eine dynamische sowie statische Zielverfolgung beinhaltet. Dabei fährt das Zielfahrzeug mit variierender Relativgeschwindigkeit stets mit einem Azimutwinkel von $\alpha = 0$ vor dem Egofahrzeug, wie in der folgenden Abbildung dargestellt ist.

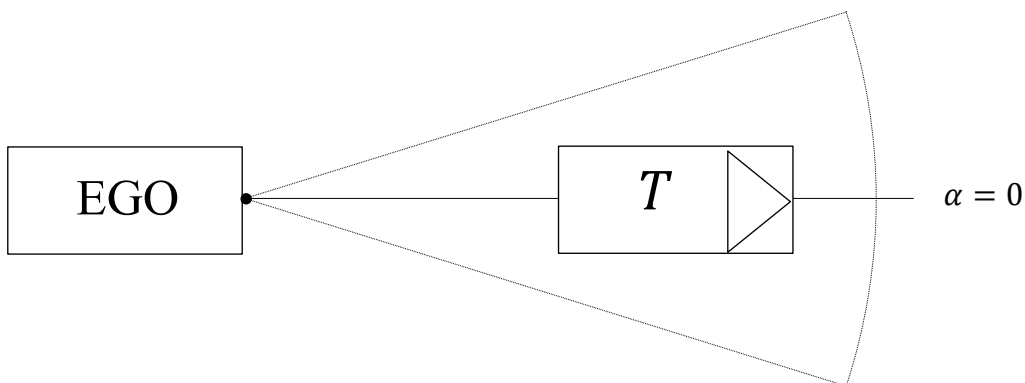


Abbildung 7-18: Szenario zur Evaluierung der Track-Kontinuität

Der zeitliche Verlauf der longitudinalen Relativgeschwindigkeit ist in Abbildung 7-19 dargestellt. Darin ist zu sehen, dass die longitudinale Relativgeschwindigkeit variiert, sodass sich diese im positiven sowie negativen Bereich befindet. Ab einem Zyklus von etwa $t_{\text{mess}} = 240$ kommen das Ziel- und Egofahrzeug beide zum Stehen, sodass eine Relativgeschwindigkeit von $v_{\text{rel},x} = 0$ vorhanden ist, bis das Zielfahrzeug ab etwa $t_{\text{mess}} = 310$ wieder beschleunigt. Das Szenario ist damit in drei Phasen unterteilt. Ziel dieses Szenarios, ist über alle Phasen hinweg eine kontinuierliche Objektverfolgung abzulegen ohne Wechsel der Track-ID.

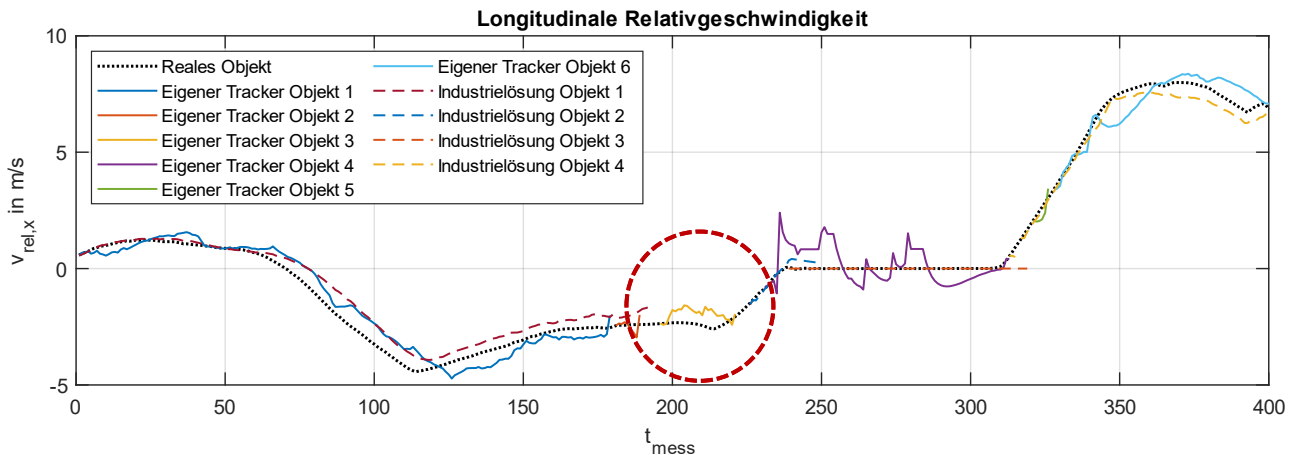


Abbildung 7-19: Vergleich der Trackfragmentierung eines verfolgten Objekts anhand der longitudinalen Geschwindigkeitsschätzung

Zusätzlich zur realen Relativgeschwindigkeit ist in der Abbildung die Geschwindigkeitsschätzung beider Tracker vorhanden. Für jedes erzeugte Objekt wird die Schätzung als eigenes Trackfragment angegeben, welches der Legende zu entnehmen ist. Bereits auf den ersten Blick ist zu erkennen, dass weder für die Industrielösung noch für den eigenen Tracker eine kontinuierliche Objektverfolgung erfolgte. Während die Industrielösung auf insgesamt vier verfolgte Objekte kommt, liegt diese Zahl für den eigenen Tracker bei sechs. Für die Industrielösung ist zusätzlich zu erkennen, dass dieser Tracker größere Lücken zwischen den verfolgten Objekten erzeugt, wie beispielsweise zwischen $t_{mess} = 180$ und $t_{mess} = 225$. In diesem Abschnitt erfolgt die Objektverfolgung nur durch den eigenen Tracker (rote Kreismarkierung). Ein weiterer Unterschied ist in der Verfolgung des statischen Objekts zwischen $t_{mess} = 235$ und $t_{mess} = 310$ zu erkennen. Die geschätzte Geschwindigkeit durch die Industrielösung liegt für statisch klassifizierte Objekte fest bei $v_{rel,x} = 0$. Eine tatsächliche Schätzung findet nicht statt, wie es beim eigenen Tracker der Fall ist. Allein aufgrund der Klassifizierung als statisches Objekt, welches nur innerhalb der Industrielösung erfolgt, findet ein Wechsel der Objektbezeichnung bzw. der Track-ID statt, da es dazu zwei unterschiedliche Objektlisten gibt. Daraus folgt, dass die Industrielösung zwar weniger Objekte verfolgt, allerdings auch größere Lücken zwischen den Trackfragmenten liegen. Zusätzlich ist eine Fragmentierung in der Verfolgung durch die Industrielösung aufgrund eines statischen Objekts nicht zu umgehen, während die kontinuierliche Verfolgung eines Objekts, welches zwischen einem dynamischen und statischen Zustand wechselt, im eigenen Tracker möglich ist.

7.3 Vergleich der Ergebnisse zwischen Messung und Simulation

Um die Übertragbarkeit der Tracking-Ergebnisse von Messung und Simulationen zu identifizieren, wurden aus vorhandenen Messdaten die Trajektorien des Zielfahrzeugs extrahiert und so konvertiert, dass diese in der Simulationssoftware als Trajektorie vorgegeben wurden. Auf diese Weise werden Detektionen des realen Radarsensors sowie des Radarsensormodells durch das gleiche Szenario erzeugt. Da das Tracking-Ergebnis, insbesondere das Clustering und die Filterung, hauptsächlich auf den Geschwindigkeiten und Positionen der vorliegenden Radardetektionen erfolgt, werden diese Größen der Messung und Simulation gegenübergestellt. Dazu folgt der Vergleich der Ergebnisse der

Geschwindigkeits-, der Positions- sowie der Gierwinkelschätzung. Zusätzlich erfolgt eine Gegenüberstellung der Trennfähigkeiten auf Basis beider Eingangsdaten.

In Abbildung 7-20 sind die Ergebnisse der Geschwindigkeitsschätzungen auf Basis von Mess- und Simulationsdaten über den zeitlichen Verlauf dargestellt. Als Referenzen sind zusätzlich die tatsächlichen Geschwindigkeiten des realen Objektes aus den Messdaten und des Simulationsobjekt aus den Simulationsdaten vorhanden. Das Zielfahrzeug fährt dabei mit einem Azimutwinkel von $\alpha = 0$ vor dem Egofahrzeug und variiert laufend seine longitudinale Relativgeschwindigkeit $v_{\text{rel},x}$. Die laterale Relativgeschwindigkeit liegt bei $v_{\text{rel},y} = 0$. Die tatsächliche Geschwindigkeit in beiden Datentypen unterscheidet sich nur minimal, sodass ein direkter Vergleich beider Tracking-Ergebnisse zulässig ist. Der Abbildung ist zu entnehmen, dass für beide Geschwindigkeitskomponenten $v_{\text{rel},x}$ (oben) und $v_{\text{rel},y}$ (unten) die Geschwindigkeitsschätzungen sich nur minimal unterscheiden. Dabei ist zu erkennen, dass über den gesamten Simulationsverlauf nicht nur ein Objekt verfolgt wird, sondern der Tracker vereinzelt ein zweites Objekt im unmittelbaren Umkreis des Zielfahrzeugs initiiert und kurzzeitig verfolgt. Diese Ausschläge sind in der longitudinalen sowie lateralen Geschwindigkeitsschätzung zu erkennen (rote Kreismarkierungen). Der Grund für diese weitere Initiierung sind zusätzliche Detektionen am Zielfahrzeug mit einer stark abweichenden Dopplergeschwindigkeit als der realen Dopplergeschwindigkeit. Diese sind nicht kontinuierlich vorhanden, sodass das zusätzliche Objekt fortlaufend initiiert und gelöscht wird. Die genaue Ursache dieser Detektionen in den Simulationsdaten wurde innerhalb dieser Arbeit aus zeitlichen Gründen nicht eindeutig geklärt und ist für eine Weiterentwicklung des Tracking-Algorithmus zu untersuchen.

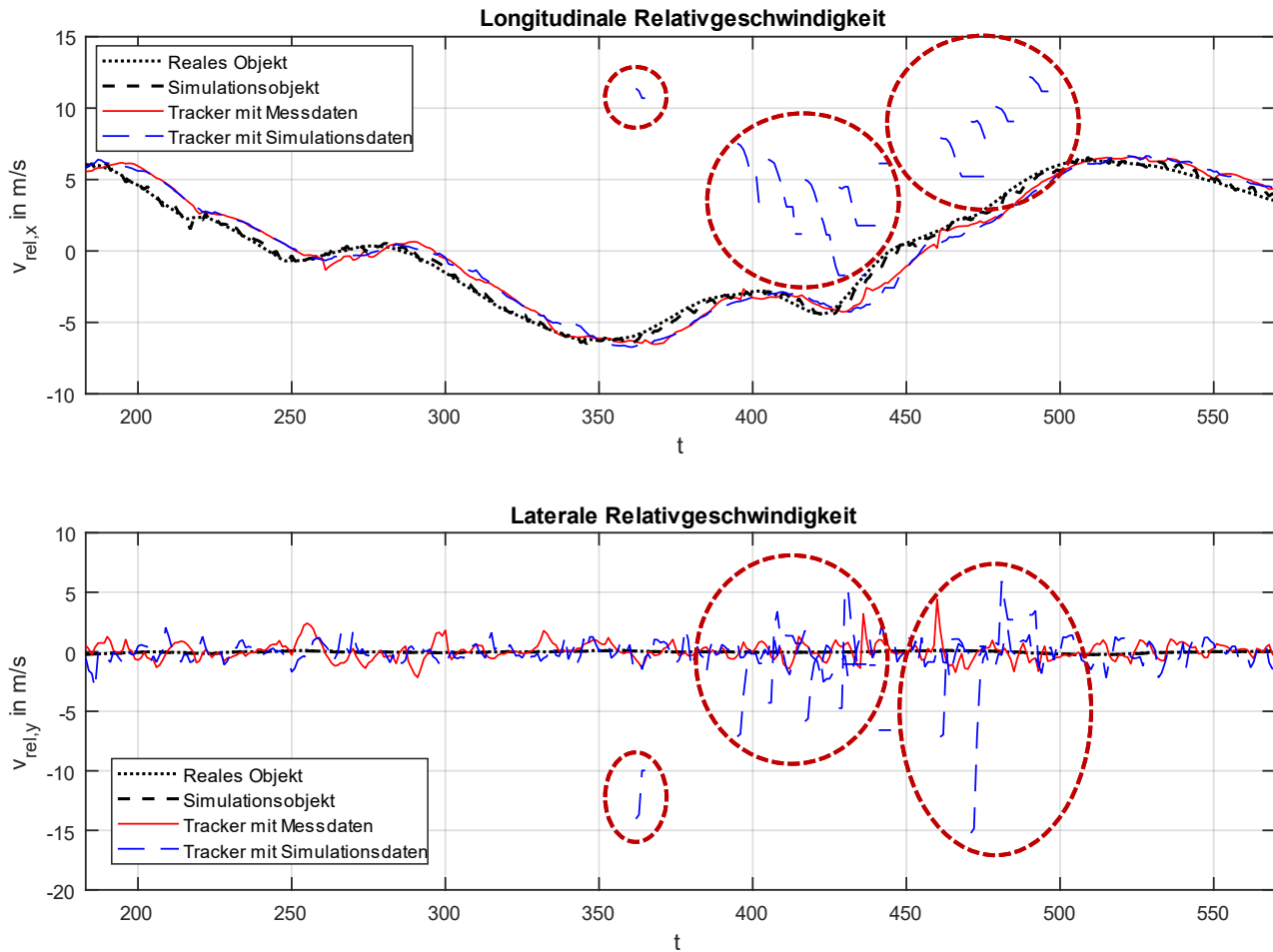


Abbildung 7-20: Vergleich der Geschwindigkeitsschätzung zu Mess- sowie Simulationsdaten

Um die Geschwindigkeitskomponenten nicht nur separat zu betrachten, wird zusätzlich der Gierwinkel beider Tracking-Ergebnisse untersucht. Dazu wird das Slalomszenario, welches in Kapitel 7.2.4 vorgestellt wurde, genutzt. In Abbildung 7-21 (oben) ist der Vergleich der Gierwinkelschätzung über den zeitlichen Verlauf dargestellt. Die tatsächlichen Gierwinkel φ des realen Objekts sowie des Simulationsobjekts stimmen unter Vernachlässigung von vereinzelt, minimalen Abweichungen überein. Ebenso stimmen die Gierwinkelschätzungen auf Basis beider Datentypen annähernd überein. Vereinzelt ist die Schätzung auf Basis von Simulationsdaten weniger träge, welches in den Zyklen 800-900 und 1000-1100 zu erkennen ist. Das lässt sich ebenfalls im Vergleich der Differenzen in der Abbildung (unten) daran erkennen, dass in diesen Intervallen die größte Amplitude der Differenz auf Basis von Simulationsdaten geringer ist. Unterscheidungen, welche das Tracking-Ergebnis beeinflussen, treten allerdings nicht auf.

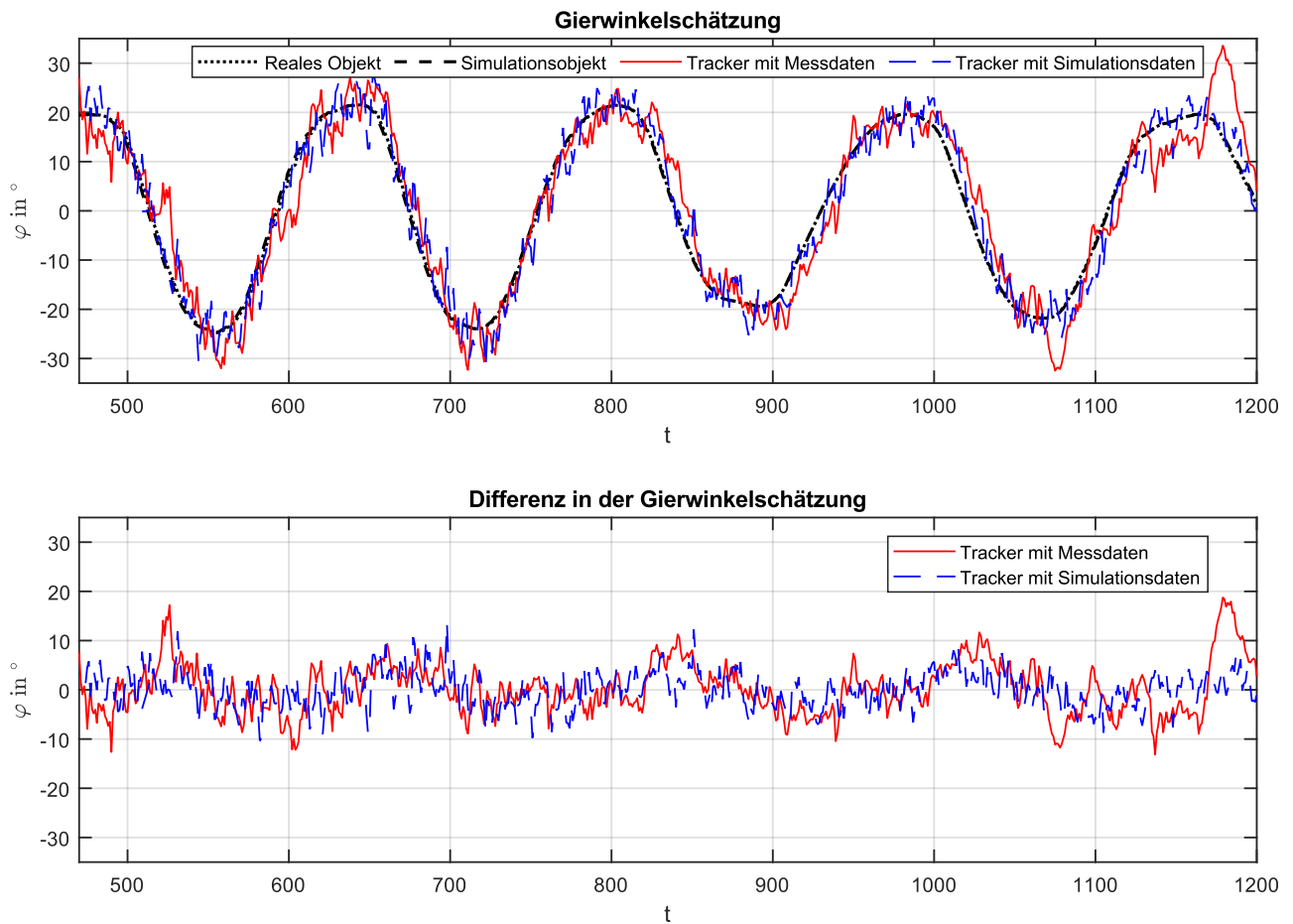


Abbildung 7-21: Vergleich der Gierwinkelschätzung zu Mess- sowie Simulationsdaten

Neben der Geschwindigkeitsschätzung beeinflusst die Positionsschätzung das Tracking-Ergebnis. In Abbildung 7-22 sind daher die Ergebnisse einer Positionsschätzung für beide Datentypen dargestellt. Im linken Diagramm ist die Trajektorie des realen und Simulationsobjekt abgebildet, während im rechten Diagramm die Differenz in der lateralen Positionsschätzung über die longitudinale Entfernung aufgetragen ist. Die laterale Positionsschätzung wird für dieses Szenario hervorgehoben, da diese entscheidend ist für die Fahrstreifenzurordnung und damit einer möglichen Trajektorienplanung des Egofahrzeuges, welches sich auf dem äußeren Fahrstreifen befindet. Die Trajektorie des realen Objekts und des Simulationsobjekts stimmen hier vollständig überein. Die Positionsschätzung auf Basis der Simulationsdaten ist hier genauer, insbesondere im Kurvenbereich. Die erzeugte Differenz liegt im Kurvenbereich bei $\Delta y = -1,25$ m und ist damit betragsmäßig über einen Meter kleiner als die Differenz auf Basis der Messdaten. Jedoch wird auch im Tracking-Verlauf der Simulationsdaten der Egofahrstreifen vereinzelt als teilweise belegt angenommen, da der Fehler in der Positionsschätzung dort zu groß wird.

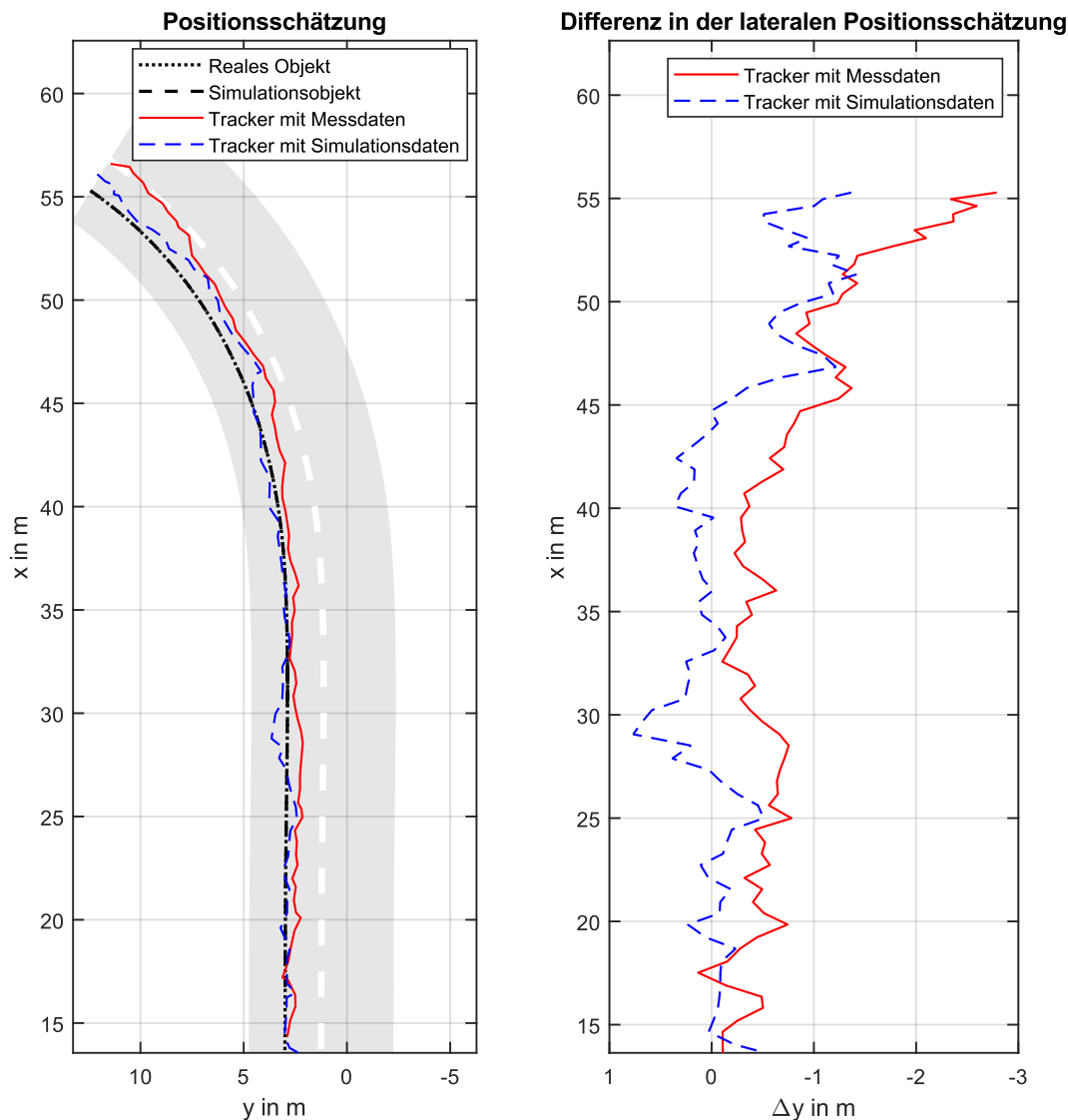


Abbildung 7-22: Vergleich der Positionsschätzung zu Mess- sowie Simulationsdaten

Die Positions- und Geschwindigkeitsschätzung beeinflussen das Tracking-Ergebnis unter anderem durch die darauf basierende Trennfähigkeit des Trackers. Dazu werden erneut die Szenarien betrachtet, welche in Kapitel 7.2.6 vorgestellt wurden. Es wird untersucht, ab welcher lateralen Distanz und ab welcher Geschwindigkeitsdifferenz zwei Zielfahrzeuge als einzelne Objekte verfolgt werden. Dazu sind in Abbildung 7-23 die Positionsschätzung auf Basis beider Datentypen dargestellt. Aus Gründen der Übersichtlichkeit wird hier auf die Abbildung der tatsächlichen Positionen beider Datentypen verzichtet. Diese wurden abgeglichen und stimmen überein, sodass nur die Positionen des realen Objekts zu sehen sind. Es ist zu erkennen, dass für beide Datentypen bei einer lateralen Distanz von $\Delta y = 3 \text{ m}$ nur ein Objekt verfolgt wird. Die Zielfahrzeuge werden demnach nicht getrennt. Eine Trennung der Objekte findet bei den Messdaten erst ab $\Delta y = 5 \text{ m}$, bei den Simulationsdaten ab $\Delta y = 5,6 \text{ m}$ statt. Die Trennfähigkeit bei beiden Datentypen befindet sich damit in der gleichen Größenordnung, wie an dem identifizierten Wert in Kapitel 7.2.6 ($\Delta y = 5,8 \text{ m}$) zu erkennen ist.

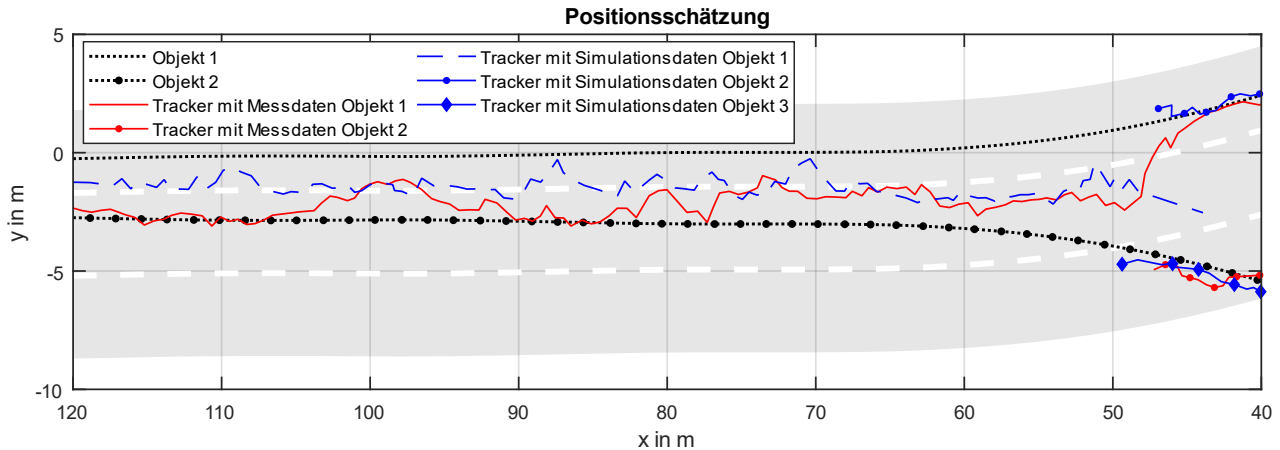


Abbildung 7-23: Vergleich der Trennfähigkeit mit Variation der lateralen Distanz zu Mess- sowie Simulationsdaten

Neben der Trennfähigkeit anhand der Distanz wird die Trennfähigkeit anhand der Geschwindigkeitsdifferenz für beide Datentypen untersucht. Dazu sind in Abbildung 7-24 die Ergebnisse der Geschwindigkeitsschätzungen als zeitlicher Verlauf dargestellt. Hier ist eine deutliche Unterscheidung beider Tracking-Ergebnisse zu finden. Während der Tracker auf Basis der Messdaten bis etwa $t = 450$ beide Zielfahrzeuge als nur ein Objekt verfolgt, zeigt der Tracker auf Basis der Simulationsdaten hier ein anderes Verhalten. Jede separate Linie steht hier für ein initiiertes und kurzzeitig verfolgtes Objekt. Dabei liegen die zugeordneten Geschwindigkeiten zwischen $v_{\text{rel},x} = -1 \frac{\text{m}}{\text{s}}$ und $v_{\text{rel},x} = 21 \frac{\text{m}}{\text{s}}$, wie in der Abbildung zu erkennen ist. Eine kontinuierliche Objektverfolgung findet bis dahin nicht statt. Ab $t = 450$ ist der Abbildung zu entnehmen, dass nun zwei Objekte getrennt verfolgt werden, allerdings zusätzlich weitere Objektinitiiierungen und kurzzeitige -verfolgungen vorkommen.

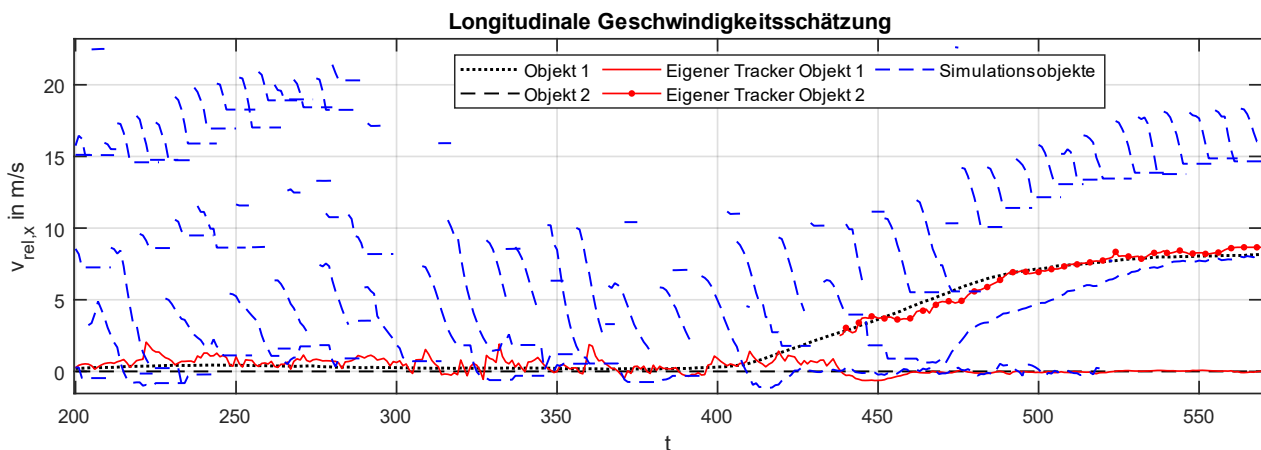


Abbildung 7-24: Vergleich der Trennfähigkeit mit Variation der Relativgeschwindigkeit zu Mess- sowie Simulationsdaten

Dieses Verhalten des Trackers auf Basis der Simulationsdaten ist auf Unterschiede in den zugewiesenen Dopplergeschwindigkeiten zu den Detektionen zurückzuführen. In Abbildung 7-25 sind dazu die Dopplergeschwindigkeiten zum einen der Messdaten und zum anderen der Simulationsdaten dargestellt. Während die Dopplergeschwindigkeiten in den Messdaten durch den Verlauf bei etwa

$\dot{r} = 0$ und dem anschließenden Anstieg plausibel sind, scheinen die Simulationsdaten einen Großteil an Detektionen zu enthalten, deren Ursprung zwar die Zielfahrzeuge sind, die zugewiesenen Dopplergeschwindigkeiten allerdings nicht mit der tatsächlichen radialen Relativgeschwindigkeit übereinstimmt. Der Ursprung der Detektionen kann zum einen an der Position, welche hier nicht dargestellt ist, sowie am qualitativen Verlauf, welcher mit der realen Geschwindigkeit insbesondere ab $t = 450$ übereinstimmt, festgemacht werden. Die Egogeschwindigkeit beträgt $v_{\text{ego}} = 10 \frac{\text{m}}{\text{s}}$, sodass die Werte bei $\dot{r} = -10 \frac{\text{m}}{\text{s}}$ durch Detektionen statischer Objekte erzeugt werden (Straßenreflexionen, Vegetation etc.). Zusätzlich ist zu erkennen, dass ab $t = 450$ die Differenz beider „Detektionsstränge“ der Egogeschwindigkeit entspricht (rote Kreismarkierung). Die Ursache dieser Dopplergeschwindigkeiten wurde aus zeitlichen Gründen innerhalb dieser Arbeit nicht eindeutig geklärt und ist für eine Weiterentwicklung des Tracking-Algorithmus zu untersuchen.

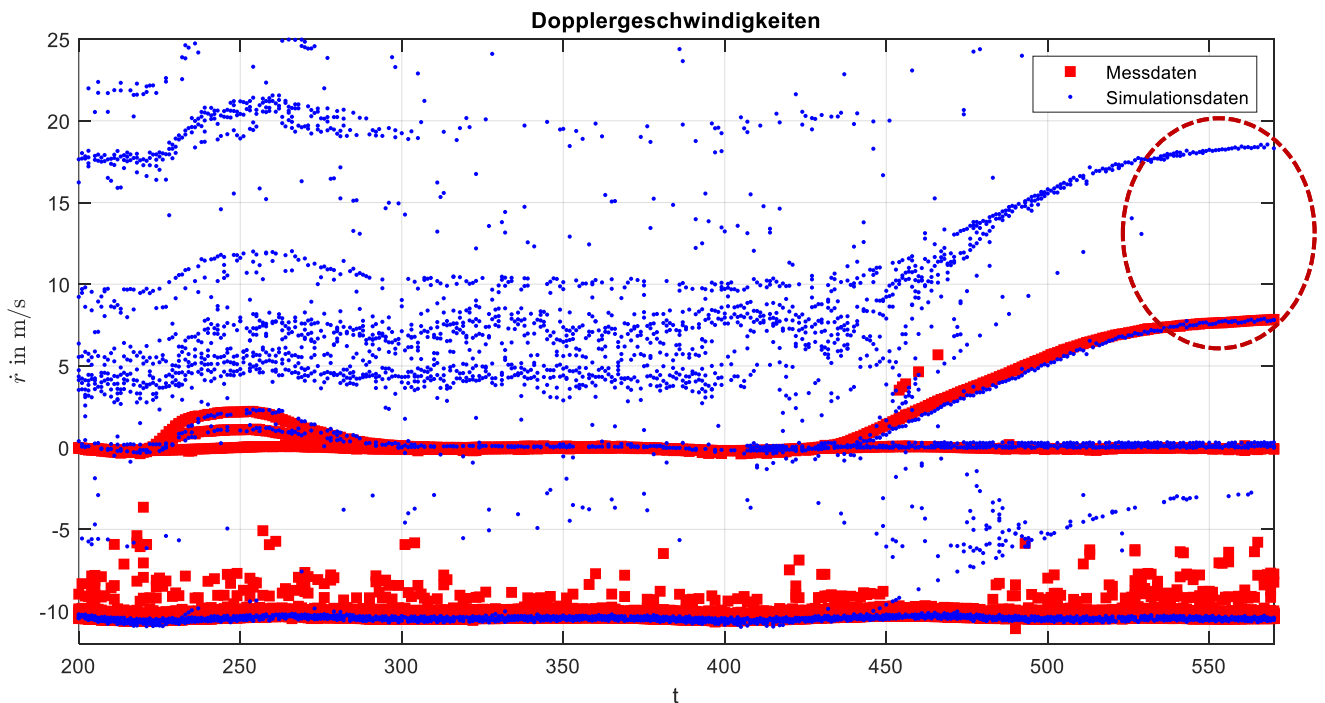


Abbildung 7-25: Vergleich der Dopplergeschwindigkeiten des realen Radarsensor sowie des Radarsensormodells

Abschließend ist zusammenzufassen, dass die Ergebnisse des Trackers von Simulations- und Messdaten grundsätzlich übertragbar sind, sofern sich die Detektionen beider Datentypen nicht schwerwiegend unterscheiden.

7.4 Zusammenfassung

Dieses Kapitel betrachtet die Evaluierung des Tracking-Algorithmus. Dazu wurde die Leistungsfähigkeit des Trackers mithilfe von Messdaten mit einer am FZD vorliegenden Industrielösung verglichen, indem zeitliche Kriterien des Track-Management, die Geschwindigkeits-, Positions- und Gierwinkelschätzung sowie die Trennfähigkeit beider Tracker untersucht wurden. Eine Bewertung der Ergebnisse zu den zeitlichen Kriterien der Trackinitiiierung und -löschung kann aufgrund des

Zielkonflikts zur Vermeidung von sowohl False-Positives als auch False-Negatives nur mit Blick auf das Anwendungsgebiet und den dort auftretenden Geschwindigkeiten getroffen werden. Zusätzlich ist der zugewiesene Status der Objekte in der Industrielösung nicht bekannt, sodass keine Unterscheidung zwischen „initiiert“ oder „bestätigt“ stattfindet. Es steht fest, dass die erkannten Objekte durch die Industrielösung grundsätzlich bereits zu einem früheren Zyklus verfolgt werden, als der eigene implementierte Tracker dieser als bestätigte Objekte einstuft. Diese Eigenschaft lässt sich aber nur mit Blick auf das Anwendungsgebiet des Tracking-Algorithmus bewerten. Befindet sich das Egofahrzeug in einer Umgebung wie beispielsweise auf einer Autobahn ist die frühe Erkennung von Objekten aufgrund der hohen Geschwindigkeiten vorteilhafter. Führt das Egofahrzeug allerdings innerorts auf langsam befahrenen Straßen, wird durch eine spätere Erkennung von Objekten eine geringere Zahl von False-Positives angestrebt. Die False-Positives wären in diesem Fall Störsignale, welche als Objekt klassifiziert und verfolgt werden. Daher kann ohne Relation zum Anwendungsgebiet innerhalb dieser Arbeit keine wertende Aussage zu den zeitlichen Kriterien beider Tracking-Algorithmen getroffen werden.

Zum Vergleich der Geschwindigkeitsschätzung beider Tracker wurden Szenarien mit annähernd konstanten Geschwindigkeiten und auch beschleunigten Bewegungen untersucht. Hier wurde deutlich, dass der implementierte Tracker grundsätzlich einen höheren Fehler erzeugt als die Industrielösung. Dieser Fehler wird durch starke Schwankungen in den Schätzungen und der Trägheit des Trackers ausgelöst. Die Industrielösung gibt stabilere Verläufe in der Schätzung der Geschwindigkeiten und verursacht durch eine geringere Trägheit insbesondere bei beschleunigten Bewegungen einen geringeren Fehler. Grundsätzlich steigt der Fehler in der Schätzung mit der Beschleunigung des Zielfahrzeuges in beiden Trackern. Zusätzlich ist die laterale Relativgeschwindigkeit bei beiden Trackern mit einem höheren Fehler belastet als die longitudinale. Im eigenen Tracker liegt die Ursache dafür in den Berechnungsmethoden der kartesischen Geschwindigkeiten. Die laterale Geschwindigkeit ist hier bei Ungenauigkeiten in den Eingangsdaten der Berechnungsmethoden fehleranfälliger als die longitudinale Geschwindigkeit.

Ähnliches Verhalten ist im Vergleich der Positionsschätzung beider Tracker festzustellen. Dazu wurden die longitudinale und laterale Schätzung zuerst separat betrachtet. Der Verlauf des eigenen Trackers ist instabiler als die Industrielösung für beide Richtungen. Allerdings bedeutet dies, trotz des höheren Fehlers in der Geschwindigkeitsschätzung, nicht notwendigerweise einen größeren Fehler in der Position. Insbesondere bei konstanten Geschwindigkeiten ist die Positionsschätzung des eigenen Trackers genauer. Nach der separaten Betrachtung wurde ein Kurvenszenario untersucht, in dem das Zielfahrzeug auf dem inneren Fahrstreifen dem Egofahrzeug entgegenkommt. Die laterale Positionsschätzung beider Tracker war an dieser Stelle nicht genau genug, um eine eindeutige Fahrstreifenzuweisung des Zielfahrzeugs zu gewährleisten. Der Egofahrstreifen wurde damit nicht als komplett frei angenommen.

Die Untersuchung der Gierwinkelschätzung erfolgte mithilfe einer Slalomfahrt des Zielfahrzeuges. Ebenfalls fällt hier ein instabilerer Verlauf des eigenen Trackers zum Gierwinkel auf. Allerdings erzeugt die Trägheit Industrielösung einen periodischen Fehler, welcher bis zu 20° groß wird, während der Fehler des implementierten Trackers hier lediglich auf bis zu 10° steigt.

Unterschiedliches Verhalten findet sich ebenfalls in der Schätzung der Bounding-Boxen wieder. Während die Industrielösung keine Mindestmaße der Objekte und damit der Bounding-Box annimmt, arbeitet der eigene Tracker mit diesen. Dies hat zur Folge, dass die verfolgten Objekte durch die Industrielösung grundsätzlich unterdimensioniert sind. Die Konsequenz der Verwendung von Mindestmaßen ist allerdings eine mögliche Überdimensionierung von Objekten wie Fahrradfahrer oder Fußgänger, welches im Sinne einer erhöhten Fahrsicherheit nicht grundsätzlich negativ zu bewerten ist. Ein weiteres Defizit des eigenen Trackers sind hier Fehler in der Gierwinkelschätzung, welche die Orientierung der Bounding-Box beeinflussen. Durch eine fehlerhafte Orientierung in Kombination mit den definierten Mindestmaßen der Bounding-Box, kann nicht nur der Fahrstreifen des Zielfahrzeug, sondern auch der Egofahrstreifen fälschlicherweise als belegt angenommen werden. Ebenso wurde basierend auf der Positions- und Geschwindigkeitsschätzung die Trennfähigkeiten beider Tracker verglichen. Während für die Trennung zweier Objekte beim implementierten Tracker eine geringere Geschwindigkeitsdifferenz nötig ist, trennt die Industrielösung zwei Objekte bei einer geringeren Distanz.

Als letztes Kriterium betrachtet die Evaluierung die Kontinuität der Objektverfolgung. Dazu wurde ein Szenario gewählt, dessen Zielfahrzeug zunächst eine gewisse Relativgeschwindigkeit aufweist und anschließend gleich dem Egofahrzeug kurzzeitig zum Stehen kommt, bevor es wieder beschleunigt. Weder der eigene Tracker noch die Industrielösung haben das Zielfahrzeug als ein Objekt kontinuierlich verfolgt. Es ist bekannt, dass die Industrielösung grundsätzlich zwischen statischen und dynamischen Objekten unterscheidet, da je Objekttyp eine Objektliste ausgegeben wird. Daher ist eine kontinuierliche Verfolgung für Szenarien dieser Art durch die Industrielösung nicht möglich, während mit einer genaueren Positions- und Geschwindigkeitsbestimmung eine kontinuierliche Verfolgung durch den eigenen Tracker möglich ist.

Zusammenfassend lässt sich damit sagen, dass die Industrielösung stabilere Schätzungen der Objektzustände liefert, diese aber insbesondere bezüglich der Positionsbestimmung und der Schätzung der Objektdimensionen nicht notwendigerweise genauer sind. Des Weiteren erfolgte die Untersuchung der Übertragbarkeit der Tracking-Ergebnisse zwischen Simulations- und Messdaten. Die Übertragbarkeit ist gewährleistet, sofern sich die Detektionen des realen Radarsensors und des Radarsensormodells nicht grundlegend unterscheiden.

8 Fazit und Ausblick

In dieser Arbeit ist ein Tracking-Algorithmus zur Verarbeitung von Radarsensordaten entwickelt worden. Der Tracker ermöglicht die zeitliche Nachverfolgung von Einzeldetektionen und gibt für sowohl synthetische als auch reale Radardaten eine Objektliste aus. Zur Datenassoziation der zusammengefassten Detektionen wird die Methode des Global Nearest Neighbor eingesetzt, die Prädiktion und Filterung übernimmt ein Extended Kalman Filter.

Die synthetischen Radardaten werden von einem Radarsensormodell, welches am Fachgebiet für Fahrzeugtechnik der Technischen Universität Darmstadt entwickelt wurde, geliefert. Am Ausgang dieses Modells liegen Radardaten vor, die sich den Daten eines realen Radars nach den Fouriertransformationen annähern. Damit geben diese Auskunft zur Entfernung, zum Azimutwinkel und zur radialen Relativgeschwindigkeit der Detektionen. Am Eingang des Tracking-Algorithmus liegen diese Radardaten als Bin-Werte vor. Die erhaltenen Detektionen werden vom Tracking-Algorithmus zunächst anhand der Empfangsleistung gefiltert, um Störsignale zu eliminieren. Anschließend findet eine Transformation der Entfernung und des Azimutwinkels in kartesische Koordinaten statt. Zusätzlich erfolgt die Bestimmung der Dopplergeschwindigkeiten unter simultaner Betrachtung von fünf Zyklen zur Auflösung der Mehrdeutigkeiten. Diese Betrachtung von zukünftigen Zyklen verursacht einen Verzug im gesamten Tracking-Algorithmus von zwei Zyklen. Zusammengefasste Mehrfachreflexionen eines Objekts werden als Cluster vorhandenen Objekten mit Hilfe des Globals Nearest Neighbor, welches die Distanz und Geschwindigkeitsdifferenz zwischen Cluster und Objektprädiktion betrachtet, zugeordnet. Die Korrektur der prädizierten Objektzustände erfolgt anschließend mit einem Extended Kalman Filter, welches den Objektzustand des nächsten Zyklus durch ein Constant Velocity Model prädiziert. Für die initiale Bestimmung der kartesischen Geschwindigkeiten werden zwei unterschiedliche Methoden eingesetzt. Die Methode zur Instantaneous Velocity Estimation oder die Berechnung auf Basis des Gierwinkels, welcher anhand der Positionsänderung geschätzt wird. Die Auswahl der Methode basiert auf unterschiedlichen Kriterien.

Die Evaluierung des implementierten Trackers erfolgte mithilfe von realen Messdaten im Vergleich zur Industrielösung, welche im vorliegenden realen Radarsensor als Tracking-Modul integriert ist. Dazu wurden zeitliche Kriterien des Track-Management, die Geschwindigkeits-, Positions- und Gierwinkelschätzung sowie die Trennfähigkeit beider Tracker untersucht. Eine Bewertung der Ergebnisse zu den zeitlichen Kriterien der Track-Initiierung und Track-Löschung kann aufgrund des Zielkonflikts zur Vermeidung von sowohl False-Positives als auch False-Negatives nur mit Blick auf das Anwendungsgebiet und den dort auftretenden Geschwindigkeiten getroffen werden. Im Vergleich der Geschwindigkeits- und Positionsschätzung wurde deutlich, dass der innerhalb dieser Arbeit implementierte Tracker gegenüber der Industrielösung einen grundsätzlich höheren Fehler erzeugt. Dieser wird durch stärkere Schwankungen in der Schätzung und der Trägheit des Trackers verursacht. Die Industrielösung läuft damit stabiler. Allerdings erzeugt auch dieser bei der Positionsschätzung einen lateralen Fehler, welcher groß genug ist, um in einer Kurvenfahrt eine eindeutige Fahrstreifenzuweisung des Zielfahrzeugs zu verhindern. Damit wird der Egofahrstreifen des Egofahrzeuges als teilweise belegt angenommen. Eine Ausnahme ist hier die Positionsschätzung eines Zielfahrzeuges mit

annähernd konstanter Geschwindigkeit, welche durch den implementierten Tracking-Algorithmus gegenüber der Industrielösung genauer ist. Grundsätzlich ist zu sagen, dass die erzeugten Fehler in der Geschwindigkeits- und Positionsschätzung bei beiden Trackern mit höherer Beschleunigung des Zielfahrzeuges steigen. Die Gierwinkelschätzung des implementierten Trackers weist ebenfalls einen geringeren Fehler gegenüber der Industrielösung auf. Dies ist auf die größere Trägheit der Industrielösung in der Schätzung dieser Größe zurückzuführen, welche einen periodischen Fehler von bis zu 20° erzeugt, während der Fehler des implementierten Trackers nur auf bis zu 10° steigt. Die untersuchte Trennfähigkeit beider Tracker weist ebenfalls Unterschiede auf. Während für die Trennung zweier Objekte beim implementierten Tracker eine geringere Geschwindigkeitsdifferenz nötig ist, trennt die Industrielösung zwei Objekte bei einer geringeren Distanz. Zusammenfassend ist mit den Ergebnissen der Evaluierung zu sagen, dass die Industrielösung stabilere Schätzungen der Objektzustände liefert, diese aber insbesondere bezüglich der Positionsbestimmung und der Schätzung der Objektdimensionen nicht grundsätzlich genauer sind.

Die Übertragbarkeit der Tracking-Ergebnisse von Simulations- und Messdaten wurde durch den Vergleich gleicher Szenarien, welche als Mess- und Simulationsdaten vorliegen, überprüft. Die Ergebnisse beider Tracking-Ergebnisse stimmen mit minimalen Abweichungen überein, insofern die realen als auch synthetischen Radardaten plausible Detektionswerte beinhalten. Die Übertragbarkeit ist damit gewährleistet, sofern sich die Detektionen des realen Radarsensors und des Radarsensormodells nicht schwerwiegend unterscheiden.

Für Weiterentwicklungen des Tracking-Algorithmus besteht die Möglichkeit den Extended Kalman Filter durch ein komplexes Interacting Multiple-Model Filter zu erweitern. Damit lassen sich mehrere Bewegungsmodelle, wie das bereits genutzte Constant Velocity Model, das Constant Acceleration Model und das Constant Turn Rate Model, vereinen und adaptiv nutzen. Zwischen den Bewegungsmodellen wird mithilfe von Übergangswahrscheinlichkeiten basierend auf der bisherigen Objektbewegung gewechselt. Eine genauere Objektverfolgung mit beschleunigten Geschwindigkeiten und bei Gierbewegungen ist damit möglich.¹²⁰ Eine weitere Anpassung des Tracking-Algorithmus stellt die Justierung der genutzten Filterparameter dar. Aus zeitlichen Gründen wurde innerhalb dieser Arbeit der getestete Wertebereich aller Parameter beschränkt. Mit einer Erweiterung der getesteten Parametervariationen und einer Feinjustierung des Filters kann die Schätzung der Objektzustände stabiler erfolgen, welches ebenfalls eine genauere Positions- und Geschwindigkeitsschätzung begünstigt.

Zur genaueren Bestimmung der kartesischen Geschwindigkeiten sind mehrere Detektionen eines Objektes innerhalb eines Zyklus notwendig. So kann die Methode der Instantaneous Velocity Estimation genutzt werden, ohne die Detektionen zweier Zyklen zu verknüpfen und dadurch Ungenauigkeiten zu verursachen. Eine größere Zahl an Detektionen eines Objektes wird durch das Herabsetzen der Leistungsschwelle erreicht, sodass mehr Empfangssignale mit niedrigerer Leistung ebenfalls als Detektion am Ausgang des Sensors vorhanden sind. Sind von einem Objekt damit stets mehrere

¹²⁰ Ming-yang, D. et al.: The interacting multiple model algorithm based on adaptive Markov transition probability (2017 - 2017).

Detektionen im gesamten Sensorfeld vorhanden, kann bereits für jedes Cluster die kartesischen Geschwindigkeiten bestimmt werden.

Eine adaptive Gestaltung des Track-Management bietet eine weitere Entwicklungsmöglichkeit. Je nach Umgebung des Egofahrzeuges variieren die benötigten Zyklen zur Bestätigung oder Löschung eines Objektes. Damit wird die Reaktionszeit für potenzielle automatisierte Manöver auf Basis des Tracking-Ergebnis den auftretenden Relativgeschwindigkeiten in der Umgebung angepasst. Für eine Anwendung im Straßenverkehr außerdem die Echtzeitfähigkeit des Tracking-Algorithmus nötig. Diese wird durch Optimierung der Rechenzeit u.a. durch Anpassungen im Code erreicht.

Mit der Erreichung einer Echtzeitfähigkeit und einer genaueren Bestimmung der Objektzustände liefert der Tracking-Algorithmus die Grundlage einer nachgestellten Trajektoriengenerierung des Egofahrzeuges. Diese wird unter anderem in autonomen Fahrzeugen genutzt, um die Manöver der aktuellen Situation, welche sich aus den verfolgten Objekten und der Egokinematik ergibt, anzupassen. Durch die Übertragbarkeit der Tracking-Ergebnisse von Simulationsdaten auf Messdaten lassen sich Erweiterungen des Tracking-Algorithmus und ebenso eine nachgestellte Trajektoriengenerierung ebenfalls mit Hilfe von synthetischen Radardaten verifizieren und validieren.

Anhang A: Parameterdatei

Tabelle 8: Genutzte Parameter innerhalb des Tracking-Algorithmus wie Schwellenwerte oder Mindestmaße

Parametername	Wert	Bedeutung
minVehicleWidthThreshold_m	1,8 m	Mindestbreite Bounding-Box
minVehicleLengthThreshold_m	4,3 m	Mindestlänge Bounding-Box
PowerRangeCompensatedThreshold_dBm3	0 dBm3	Grenzwert zur Filterung von Störsignale
DopplerThreshold_m_s	1 m/s	Grenzwert in der Dopplerberechnung (maximale Differenz zwischen „gleich“ klassifizierten Werten)
RangeRateThreshold_check	20 m/s	Grenzwert in der Dopplerberechnung: Maximale Differenz zwischen bestimmter Dopplergeschwindigkeit und Entfernungsänderung der gleich klassifizierten Werten)
maxNumTracks	100	Maximale Zahl von Objekten im selben Zyklus
ClusterThresholdDistance_m	5 m	Grenzwert zum Clustering (maximale euklidische Distanz der Detektionen)
ClusterThresholdDopplerBin	20	Grenzwert zum Clustering (maximale Differenz der Dopplerbins der Detektionen)
ClusterThresholdDopplerVelEstimation_m_s	2 m/s	Grenzwert zum Clustern (maximale Differenz zwischen Dopplergeschwindigkeiten, um diesen in Clustser zu ergänzen)
InnovationPosThreshold_m2	9 m ²	Grenzwert zur Datenassoziation (maximale euklidische Distanz zwischen Cluster und Objekt)
InnovationDopplerVelThreshold_m2_s2	9 m ²	Grenzwert zur Datenassoziation (maximale Differenz in Dopplergeschwindigkeit zwischen Cluster und Objekt)
maxTrackCountInit	5	Maximale Zahl an Zyklen in der Initiierungsphase
minTrackCountInit	3	Minimale Zahl an erfolgten Datenassoziation zur Bestätigung
maxTrackCountDelet	10	Maximale Zahl an Zyklen in der Löschphase
TrackThresholdCenter_m	2 m	Grenzwert zur Track-Löschung
TrackThresholdVrel_m_s	2 m/s	Grenzwert zur Track-Löschung

Anhang B: Visualisierung

Bei aktivierter Visualisierung wird in einem Plot-Fenster der aktuelle Stand des Tracking-Algorithmus dargestellt und mit jedem Zyklus aktualisiert, wie in der folgenden Abbildung beispielhaft zu sehen ist. Zu sehen sind außerdem die Sichtbereiche des Fern- und Nahbereichssensor. Aktuell sind allerdings nur Daten des Fernbereichssensor vorhanden. Visualisiert werden zum einen alle Detektionen des Sensors und zum anderen die verfolgten Objekte inklusive ihrer Track-ID. Die Detektionen sind jeweils punktförmig dargestellt und je nach zugehöriger abstandskompensierter Leistung eingefärbt. Die Objekte haben eine Unterschiedliche Darstellung je nach ihrem aktuellen Status. Ist ein Objekt in der Initiierungsphase wird es als graues gefülltes Quadrat visualisiert, in der Löschphase als ein blaues ungefülltes Quadrat. Ist ein Objekt bestätigt, wird die Bounding-Box in grün angezeigt.

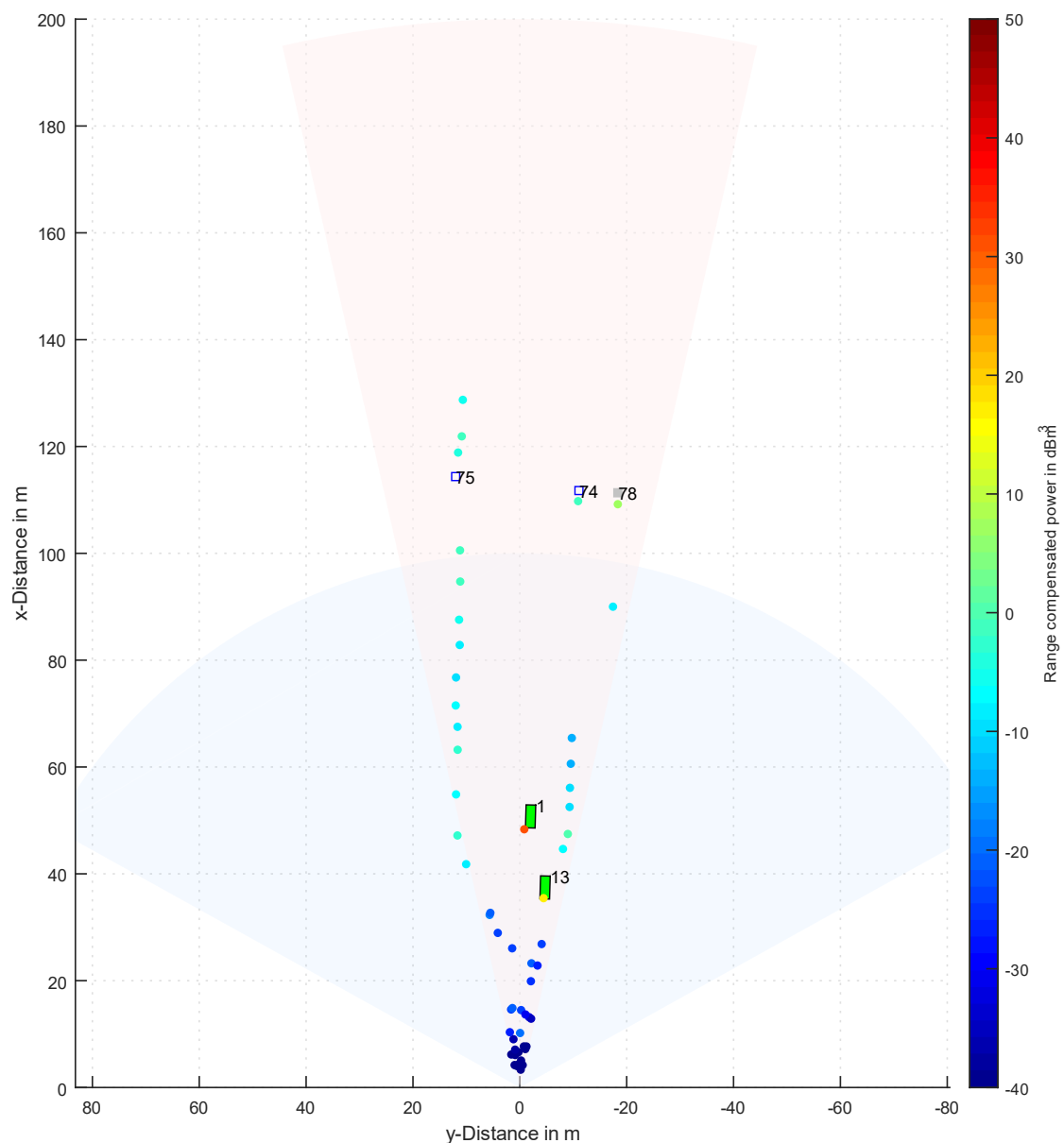


Abbildung C-1: Beispielhafte Darstellung der Visualisierung zum implementierten Tracking-Algorithmus

Anhang C: Soll- und Ist-Zeitplan

Soll-Zeitplan:

Kalenderwoche	43	44	45	46	47	48	49	50	51	52	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	April
Zwischenkolloquium: 12.12.2019																												
Abgabe Ausarbeitung: 22.04.2020																												
Abschlusskolloquium: 28.04.2020																												
Einarbeitung																												
Literaturrecherche																												
Einarbeitung VTD und Sensormodell																												
Anforderungsanalyse Algorithmus																												
Entwicklung																												
Implementierung des Prototyp																												
Implementierung des vollständigen Algorithmus																												
Definition und Implementierung von Test Cases (Simulation + Real)																												
Verifikation Algorithmus																												
Bewertung																												
Implementierung von Referenzszenarien ggf. Aufzeichnung von Realfahrten																												
Vergleich Messdaten vs. Simulationsdaten																												
Dokumentationen																												
Vonversion Ausarbeitung																												
Präsentation Zwischenkolloquium																												
Ausarbeitung																												
Präsentation																												

Ist-Zeitplan:

Kalenderwoche	43	44	45	46	47	48	49	50	51	52	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

Literaturverzeichnis

Andrew Stove, G.: Obstacle detection radar for cars (1991)

Andrew Stove, G.: Obstacle detection radar for cars, in: Electronics & Communications Engineering Journal (5), Jahrgang 3, S. 232, 1991

Audi AG: The Pulse of Autonomous Driving (2019)

Audi AG: The Pulse of Autonomous Driving; <https://www.audi.com/content/dam/gbp2/company/research/audi-beyond/2019/pulse-of-autonomous-driving/and-audi-study-autonomous-driving.pdf>, 2019, Zugriff 29.05.2020

Balzer, P.: Das Kalman-Filter (2016)

Balzer, P.: Das Kalman-Filter einfach erklärt [Teil 2]; <https://www.cbcity.de/das-kalman-filter-einfach-erklart-teil-2>, 2016, Zugriff 19.05.2020

Bar-Shalom, Y. et al.: The Probabilistic Data Association Filter (2009)

Bar-Shalom, Y.; Daum, F.; Huang, J.: The Probabilistic Data Association Filter, in: IEEE Control Systems Magazine, 2009

Bar-Shalom, Y.; Tse, E.: Tracking in a Cluttered Environment With Probabilistic Data Association (1975)

Bar-Shalom, Yaakov; Tse, Edison: Tracking in a Cluttered Environment With Probabilistic Data Association, in: Automatica (5), Jahrgang 11, S. 451–460, 1975

Beard, M. et al.: Multiple Extended Target Tracking With Labeled Random Finite Sets (2016)

Beard, Michael; Reuter, Stephan; Granstrom, Karl; Vo, Ba-Tuong; Vo, Ba-Ngu; Scheel, Alexander: Multiple Extended Target Tracking With Labeled Random Finite Sets, in: IEEE Transactions on Signal Processing (7), Jahrgang 64, S. 1638–1653, 2016

Blackman, S.; Popoli, R.: Design of tracking systems (1999)

Blackman, Samuel; Popoli, Robert: Design and analysis of modern tracking systems, Artech House Radar Library, Artech House, Boston, 1999

Blanding, W. R. et al.: Multisensor Track Management for Targets with Fluctuating SNR (2009)

Blanding, W. R.; Willett, P. K.; Bar-Shalom, Y.; Coraluppi, S.: Multisensor Track Management for Targets with Fluctuating SNR, in: IEEE Transactions on Aerospace and Electronic Systems (4), Jahrgang 45, S. 1275–1292, 2009

Broßeit, P. et al.: Shape Estimation for Extended Object Tracking

Broßeit, P.; Rapp, M.; Appenrodt, N.; Dickmann, J.: Probabilistic Rectangular-Shape Estimation for Extended Object Tracking, in: 2016 IEEE Intelligent Vehicles Symposium (IV), IEEE

Cao, Y.: Learning the Extended Kalman Filter (2008)

Cao, Yi: Learning the Extended Kalman Filter; <https://de.mathworks.com/matlabcentral/file-exchange/18189-learning-the-extended-kalman-filter>, 2008, Zugriff 30.05.2020

Challa, S. et al.: Fundamentals of Object Tracking (2011)

Challa, Subhash; Morelande, Mark R.; Musicki, Darko; Evans, Robin J.: Fundamentals of Object Tracking, Cambridge University Press, Cambridge, 2011

Clark, D. E.: Diss., Tracking with the Probability Hypothesis Density Filter (2006)

Clark, Daniel E.: Multiple Target Tracking with the Probability Hypothesis Density Filter, Dissertation Heriot-Watt University, 2006

Cobeldick, S.: Natural-Order Filename Sort (2019)

Cobeldick, Stephen: Natural-Order Filename Sort; <https://mathworks.com/matlabcentral/file-exchange/47434-natural-order-filename-sort>, 2019, Zugriff 29.05.2020

Danielsson, L.: Diss., Tracking and radar sensor modelling (2010)

Danielsson, Lars: Tracking and radar sensor modelling for automotive safety systems, Dissertation

Chalmers University of Technology, Doktorsavhandlingar vid Chalmers Tekniska Högskola N.S., 3064, Chalmers Univ. of Technology, Göteborg, 2010

Detlefsen, J.: Radartechnik (1989)

Detlefsen, Jürgen: Radartechnik, Nachrichtentechnik, Jahrgang 18, Springer Berlin Heidelberg, Berlin, Heidelberg, 1989

Dietmayer, K.: Prädiktion von maschineller Wahrnehmungsleistung beim automatisierten Fahren (2015)

Dietmayer, Klaus: Prädiktion von maschineller Wahrnehmungsleistung beim automatisierten Fahren, in: Maurer, Markus et al. (Hrsg.): Autonomes Fahren, Springer Berlin Heidelberg, Berlin, Heidelberg, 2015

Ester, M. et al.: DBSCAN for discovering clusters (1996)

Ester, M.; Kriegel, H.-P.; Sander, J.; Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise, in: AAAI Press (Hrsg.): Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD'96), 1996

Fortmann, T. et al.: Sonar tracking of multiple targets using joint probabilistic data association (1983)

Fortmann, T.; Bar-Shalom, Y.; Scheffe, M.: Sonar tracking of multiple targets using joint probabilistic data association, in: IEEE Journal of Oceanic Engineering (3), Jahrgang 8, S. 173–184, 1983

Glassner, A. S.: An Introduction to Ray tracing (1989)

Glassner, Andrew S.: An Introduction to Ray tracing, The Morgan Kaufmann Series in Computer Graphics, Elsevier Science, Saint Louis, 1989

Gosala, N. B.; Meng, X.: An RLS-Based Instantaneous Velocity Estimator for Extended Radar Tracking (2020)

Gosala, Nikhil B.; Meng, Xiaoli: An RLS-Based Instantaneous Velocity Estimator for Extended Radar Tracking, 2020

Granstrom, K. et al.: Extended Target Tracking using a Gaussian-Mixture PHD Filter (2012)

Granstrom, Karl; Lundquist, Christian; Orguner, Omut: Extended Target Tracking using a Gaussian-Mixture PHD Filter, in: IEEE Transactions on Aerospace and Electronic Systems (4), Jahrgang 48, S. 3268–3286, 2012

Granström, K. et al.: Introduction to Extended Object Tracking (2016)

Granström, Karl; Baum, Marcus; Reuter, Stephan: Extended Object Tracking: Introduction, Overview and Applications, in: Journal of Advances in Information Fusion, 2016

Granström, K.; Svensson, L.: Introduction to Single-Object Tracking (2019)

Granström, Karl; Svensson, Lennart: Multi-Object Tracking for Automotive Systems. 2.1.1 Introduction to Single-Object Tracking, edX, 2019

Hammarstrand, L. et al.: Extended Object Tracking using a Radar Resolution Model (2012)

Hammarstrand, Lars; Svensson, Lennart; Sandblom, Fredrik; Sorstedt, Joakim: Extended Object Tracking using a Radar Resolution Model, in: IEEE Transactions on Aerospace and Electronic Systems (3), Jahrgang 48, S. 2371–2386, 2012

Hu, B. et al.: Random Finite Set Based Bayesian Filtering with OpenCL in a Heterogeneous Platform (2017)

Hu, Biao; Sharif, Uzair; Koner, Rajat; Chen, Guang; Huang, Kai; Zhang, Feihu; Stechele, Walter; Knoll, Alois: Random Finite Set Based Bayesian Filtering with OpenCL in a Heterogeneous Platform, in: Sensors (Basel, Switzerland) (4), Jahrgang 17, 2017

Hyun, E.; Lee, J.-H.: Track Management Table

Hyun, E.; Lee, J.-H.: Multi-Target Tracking Scheme using a Track Management Table for Automotive Radar Systems, in: 2016 17th International Radar Symposium (IRS), IEEE

Julier, S. J.; Uhlmann, J. K.: New extension of the Kalman filter to nonlinear systems (1997)

Julier, Simon J.; Uhlmann, Jeffrey K.: New extension of the Kalman filter to nonlinear systems, in:

Kadar, Ivan (Hrsg.): Signal Processing, Sensor Fusion, and Target Recognition VI, Orlando, FL, USA, SPIE Proceedings, SPIE, 1997

Kellner, D. et al.: Grid-based DBSCAN for clustering (2012)

Kellner, Dominik; Klappstein, Jens; Dietmayer, Klaus: Grid-based DBSCAN for clustering extended objects in radar data, in: IEEE Intelligent Vehicles Symposium (IV), 2012, IEEE, Piscataway, NJ, 2012

Kim, C. et al.: Multiple Hypothesis Tracking Revisited (2015 - 2015)

Kim, Chanho; Li, Fuxin; Ciptadi, Arridhana; Rehg, James M.: Multiple Hypothesis Tracking Revisited, in: 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, IEEE, 07.12.2015 - 13.12.2015

Kim, Y.-S.; Hong, K.-S.: An IMM algorithm with federated information mode-matched filters for AGV (2007)

Kim, Yong-Shik; Hong, Keum-Shik: An IMM algorithm with federated information mode-matched filters for AGV, in: International Journal of Adaptive Control and Signal Processing (7), Jahrgang 21, S. 533–555, 2007

Kwik Team: NPY-Matlab (2018)

Kwik Team: NPY-Matlab; <https://github.com/kwikteam/np-matlab>, 2018, Zugriff 29.05.2020

Li, X. R. et al.: SPRT-Based track confirmation and rejection (2002)

Li, X. R.; Li, Ning; Jilkov, V. P.: SPRT-Based track confirmation and rejection, in: Proceedings of the Fifth International Conference on Information Fusion. FUSION 2002. (IEEE Cat.No.02EX5997), Annapolis, MD, USA, Int. Soc. Inf. Fusion, 8-11 July 2002

Linnhoff, C.: Masterthesis, Entwicklung eines Radar-Sensormodells (2018)

Linnhoff, C.: Entwicklung eines Radar-Sensormodells, Masterthesis Technische Universität Darmstadt, Darmstadt, 2018

Liu, F.: Diss., Objektverfolgung durch Sensordatenfusion (2009)

Liu, Feng: Objektverfolgung durch Fusion von Radar und Monokameradaten auf Merkmalsebene für zukünftige Fahrerassistenzsysteme, Dissertation Karlsruher Institut für Technologie, 2009

Ludloff, A.: Handbuch Radar (1993)

Ludloff, Albrecht: Handbuch Radar und Radarsignalverarbeitung, Vieweg+Teubner Verlag, Wiesbaden, 1993

Lundquist, C. et al.: An Extended CPHD Filter and GGIW Implementation (2013)

Lundquist, Christian; Granstrom, Karl; Orguner, Umut: An Extended Target CPHD Filter and a Gamma Gaussian Inverse Wishart Implementation, in: IEEE Journal of Selected Topics in Signal Processing (3), Jahrgang 7, S. 472–483, 2013

M. Kronauge et al.: Radar target detection and Doppler ambiguity resolution (2010)

M. Kronauge; C. Schroeder; H. Rohling: Radar target detection and Doppler ambiguity resolution, in: IEEE (Hrsg.): 11th International Radar Symposium, Vilnius, Litauen, I E E E, 2010

Mahler, R.: PHD filters of higher order in target number (2007)

Mahler, R.: PHD filters of higher order in target number, in: IEEE Transactions on Aerospace and Electronic Systems (4), Jahrgang 43, S. 1523–1543, 2007

Mahler, R.P.S.: Multitarget bayes filtering via first-order multitarget moments (2003)

Mahler, R.P.S.: Multitarget bayes filtering via first-order multitarget moments, in: IEEE Transactions on Aerospace and Electronic Systems (4), Jahrgang 39, S. 1152–1178, 2003

Mahler, R.P.S.: "Statistics 101" for multisensor, multitarget data fusion (2004)

Mahler, R.P.S.: "Statistics 101" for multisensor, multitarget data fusion, in: IEEE Aerospace and Electronic Systems Magazine (1), Jahrgang 19, S. 53–64, 2004

Manjunath, A. et al.: Radar Based Object Tracking (2018)

Manjunath, Ankith; Liu, Ying; Henriques, Bernardo; Engstle, Armin: Radar Based Object Detection and Tracking for Autonomous Driving, in: 2018 IEEE MTT-S International Conference on Microwaves for Intelligent Mobility (ICMIM), Munich, IEEE, 2018

Ming-yang, D. et al.: The interacting multiple model algorithm based on adaptive Markov transition probability (2017 - 2017)

Ming-yang, Du; Da-ping, Bi; Shu-liang, Wang: The interacting multiple model algorithm based on adaptive Markov transition probability, in: 2017 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC), Xiamen, IEEE, 22.10.2017 - 25.10.2017

Munz, M.: Zustands- und Existenzschätzung für die Fahrzeugumfeldererkennung (2011)

Munz, Michael: Generisches Sensorfusionsframework zur gleichzeitigen Zustands- und Existenzschätzung für die Fahrzeugumfeldererkennung, Schriftenreihe des Instituts für Mess-, Regel- und Mikrotechnik, Univ. [Fak. für Ingenieurwiss. und Informatik], Ulm, 2011

Musicki, D.; Evans, R.: Joint integrated probabilistic data association: JIPDA (2004)

Musicki, D.; Evans, R.: Joint integrated probabilistic data association: JIPDA, in: IEEE Transactions on Aerospace and Electronic Systems (3), Jahrgang 40, S. 1093–1099, 2004

P. Blom, H.: An efficient filter for abruptly changing systems (1984 - 1984)

P. Blom, H.: An efficient filter for abruptly changing systems, in: The 23rd IEEE Conference on Decision and Control, Las Vegas, Nevada, USA, IEEE, 12.12.1984 - 14.12.1984

Pietzsch, S.: Diss., Sensordatenfusion zur Erfassung komplexer Fahrzeugumgebungen (2015)

Pietzsch, Sylvia: Modellgestützte Sensordatenfusion von Laserscanner und Radar zur Erfassung komplexer Fahrzeugumgebungen, Dissertation Technische Universität München, München, 2015

Pirttilä, J. et al.: A Proposed Solution to the Range–Doppler Dilemma (2005)

Pirttilä, Juha; Lehtinen, Markku S.; Huuskonen, Asko; Markkanen, Markku: A Proposed Solution to the Range–Doppler Dilemma of Weather Radar Measurements by Using the SMPRF Codes, Practical Results, and a Comparison with Operational Measurements, in: Journal of Applied Meteorology (9), Jahrgang 44, S. 1375–1390, 2005

Rahmathullah, A. S. et al.: Generalized optimal sub-pattern assignment metric (2017)

Rahmathullah, Abu S.; García-Fernández, Ángel F.; Svensson, Lennart: Generalized optimal sub-pattern assignment metric, in: , S. 1–8, 2017

Rahmathullah, A. S. et al.: GOSPA (2018)

Rahmathullah, Abu S.; Svensson, Lennart; García-Fernández, Ángel F.: GOSPA; <https://github.com/abusajana/GOSPA>, 2018, Zugriff 29.05.2020

Reid, D.: An algorithm for tracking multiple targets (1979)

Reid, D.: An algorithm for tracking multiple targets, in: IEEE Transactions on Automatic Control (6), Jahrgang 24, S. 843–854, 1979

Reuter, S. et al.: Real-Time Multi-Object Tracking using Random Finite Sets (2013)

Reuter, Stephan; Wilking, Benjamin; Wiest, Jürgen; Munz, Michael; Dietmayer, Klaus: Real-Time Multi-Object Tracking using Random Finite Sets, in: IEEE Transactions on Aerospace and Electronic Systems (4), Jahrgang 49, S. 2666–2678, 2013

Rezende, M. C. et al.: Radar Cross Section Measurements (2002)

Rezende, M. C.; Alexandre, M.; Miacci, S.; Nohara, E. L.: Radar Cross Section Measurements (8–12 GHz) of Magnetic and Dielectric Microwave Absorbing Thin Sheets, in: Revista de Física Aplicada e Instrumentacao Vol. 15, No. 1, 2002

Richards, M. A.: Radar signal processing (2005)

Richards, M. A.: Radar signal processing, McGraw-Hill professional engineering, McGraw-Hill Professional, New York, 2005

Ristic, B. et al.: A Metric for Performance Evaluation of Multi-Target Tracking Algorithms (2011)

Ristic, B.; Vo, Ba-Ngu; Clark, D.; Vo, Ba-Tuong: A Metric for Performance Evaluation of Multi-Target Tracking Algorithms, in: IEEE Transactions on Signal Processing (7), Jahrgang 59, S. 3452–3457, 2011

Rohling, H.: Radar CFAR Thresholding in Clutter and Multiple Target Situations (1983)

Rohling, Hermann: Radar CFAR Thresholding in Clutter and Multiple Target Situations, in: IEEE Transactions on Aerospace and Electronic Systems (4)AES-19, S. 608–621, 1983

Schuster, M.: Diss., Multiple Object Tracking using JIPDA (2017)

Schuster, Michael: Multiple Object Tracking for Extended Targets using JIPDA filters, Dissertation Technische Universität Chemnitz, 2017

Skolnik, M. I.: Introduction to radar systems (1980)

Skolnik, Merrill I.: Introduction to radar systems, 2. Auflage, McGraw-Hill, New York, 1980

Stefan Ackermann: Masterthesis, Systematische Untersuchung von Radar Tracking-Algorithmen (2017)

Stefan Ackermann: Systematische Untersuchung von Radar Tracking-Algorithmen, Masterthesis Technische Universität Darmstadt, Darmstadt, 2017

Thrun, S. et al.: Probabilistic robotics (2010)

Thrun, Sebastian; Burgard, Wolfram; Fox, Dieter: Probabilistic robotics, Intelligent robotics and autonomous agents, MIT Press, Cambridge, Mass., 2010

Vo, B.-T. et al.: The Cardinality Balanced Multi-Target Multi-Bernoulli Filter and Its Implementations (2009)

Vo, Ba-Tuong; Vo, Ba-Ngu; Cantoni, A.: The Cardinality Balanced Multi-Target Multi-Bernoulli Filter and Its Implementations, in: IEEE Transactions on Signal Processing (2), Jahrgang 57, S. 409–423, 2009

Wan, E. A.; van der Merwe, R.: The unscented Kalman filter for nonlinear estimation (2000)

Wan, E. A.; van der Merwe, R.: The unscented Kalman filter for nonlinear estimation, in: IEEE 2000 adaptive systems for signal processing, communications and control symposium, Lake Louise, Alta., Canada, IEEE, 2000

Winner, H.: Radarsensorik (2015)

Winner, Hermann: Radarsensorik, in: Winner, Hermann (Hrsg.): Handbuch Fahrerassistenzsysteme, ATZ/MTZ-Fachbuch, 3. Auflage, Springer Vieweg, Wiesbaden, 2015

Wood, T. M.: Diss., Random Finite Sets (2011)

Wood, Trevor M.: Random Finite Sets for Multitarget Tracking with Applications, Dissertation University of Oxford, 2011